NeOn-project.org

Neon Toolkit User's Guide

Deliverable Co-ordinator: **Michael Erdmann**

Deliverable Co-ordinating Institution: **Ontoprise GmbH (ONTO)**

Other Authors: **Thomas Hemp (ONTO)**

This document contains the first version of the online documentation for the NeOn Toolkit. The original deliverable consists of a set of HTML pages which are integrated into the toolkit as an Eclipse Help-plug-in. This document simply reproduces the same content in linear form as a Word document.

Work package participants

The following partners have taken an active part in the work leading to the elaboration of this document, even if they might not have directly contributed writing parts of this document:

Ontoprise GmbH

Executive Summary

This document contains the first version of the online documentation for the NeOn Toolkit. The original deliverable consists of a set of HTML pages which are integrated into the toolkit as an Eclipse Help-plug-in. This document simple reproduces the same content in linear forma as a Word document.

2007 © Copyright lies with the respective authors and their institutions.

Table of Contents

1. INTRODUCTION....4

2. BASICS OF NEON TOOLKIT....5

2.1. Starting NeOn Toolkit....5
2.2. Hot Keys....6
2.3. The User Interface....7
2.4. Perspectives....8
2.5. NeOn Toolkit Menus....9
2.5.1. The „File“ Menu....9
2.5.2. The „Edit“ Menu....11
2.5.3. The „Navigate“ Menu....11
2.5.4. The „Search“ Menu....11
2.5.5. The „Window“ Menu....12
2.5.6. The „Help“ Menu....13
2.6. The Tool Bar of NeOn Toolkit....14
2.7. Importing Ontologies....15
2.7.1. Supported Import Formats....16
2.7.2. File System Import....16
2.8. Exporting Ontologies....17
2.8.1. Supported Export Filters....18
2.8.2. File System Export....18
2.9. Localization of Ontologies....18
2.10. How to exit NeOn Toolkit....19

3. ONTOLOGY MODELLING....21

3.1. Schema Perspective....21
3.2. The Ontology Navigator....22
3.3. Entity Properties View for Concepts....23
3.4. Entity Properties View for Relations....25
3.5. Entity Properties View for Attributes....26
3.6. Instance View....27
3.6.1. Defining a New Instance....27
3.6.2. Deleting Instances....28
3.6.3. Moving Instances....28
3.6.4. Properties of Instances....28
3.7. Namespace Handling....29
3.8. Synonym View....29

4. VISUALIZING ONTOLOGIES....31

4.1. Ontology Graph Visualize View....31
4.1.1. Concept in Focus....32
4.1.2. Navigation history....32
4.1.3. Legend....32
4.1.4. Graph elements....32
4.1.5. Other Functions....33
4.2. Graph View....33

5. RULES....35

5.1. Creating a Rule....35
5.2. Entity Property View for Rules....35
5.3. Textually Editing Rules....36
5.4. Graphical Rule Editor....37
5.4.1. Overview....37
5.4.2. Adding Concepts and Instances to the GRE....38

5.4.3. Adding Attributes to the GRE *38*
5.4.4. Adding Predicates to the GRE *38*
5.4.5. Adding Relations or Attributes to the GRE *38*
5.4.6. Adding a Comparison *39*
5.4.7. Specifying Head and Body Parts *39*
5.4.8. Negations *39*
5.4.9. The Palette Menu *39*
5.4.10. The Context Menu *40*
5.5. Stylized English View 41

6. SAMPLE WALKTHROUGH 42

6.1. First Steps 42
6.2. Creating a New Ontology 43
6.3. Creating New Concepts 44
6.4. Multilingual Representations 45
6.5. Creating Attributes 45
6.6. Creating Relations 46
6.7. Setting up Instances 47
6.8. Moving a Concept 47

2007 © Copyright lies with the respective authors and their institutions.

1. Introduction

This document represents the first iteration of the online documentation of the NeOn Toolkit. Since the online documentation actually consists of a set of HTML pages that are available from within the toolkit and can be accessed interactively, here we can only present a version of the content in a more *linear* form.

Each top-level section of this document will be presented as a top-level node in the table of content of the on-line help system within the NeOn Toolkit. All second and third-level sections will be represented in the on-line help as an individual web pages accessible from a built-in web browser.

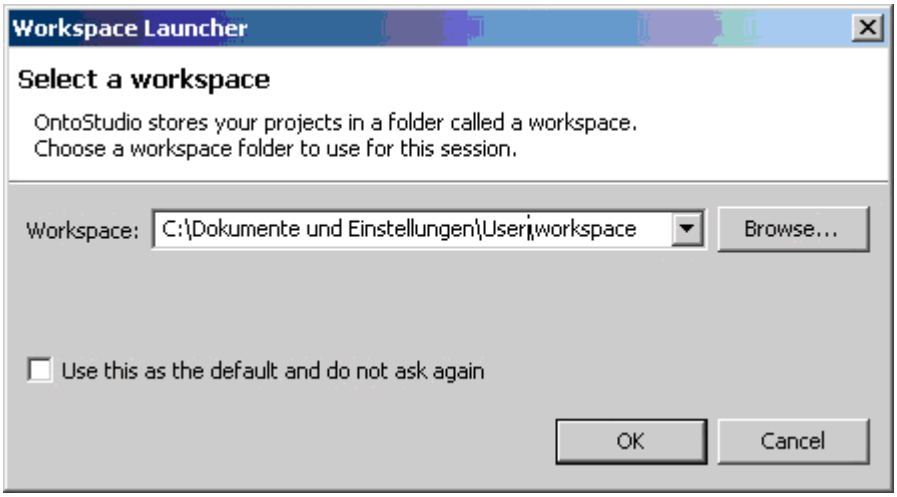
2. Basics of NeOn Toolkit

2.1. Starting NeOn Toolkit

You can start NeOn Toolkit using one of these two methods:

- Click on the NeOn Toolkit icon on the desktop or
- Select in the Windows start menu: Start > All Programs > NeOn Toolkit Version 1.0 > NeOn Toolkit 1.0

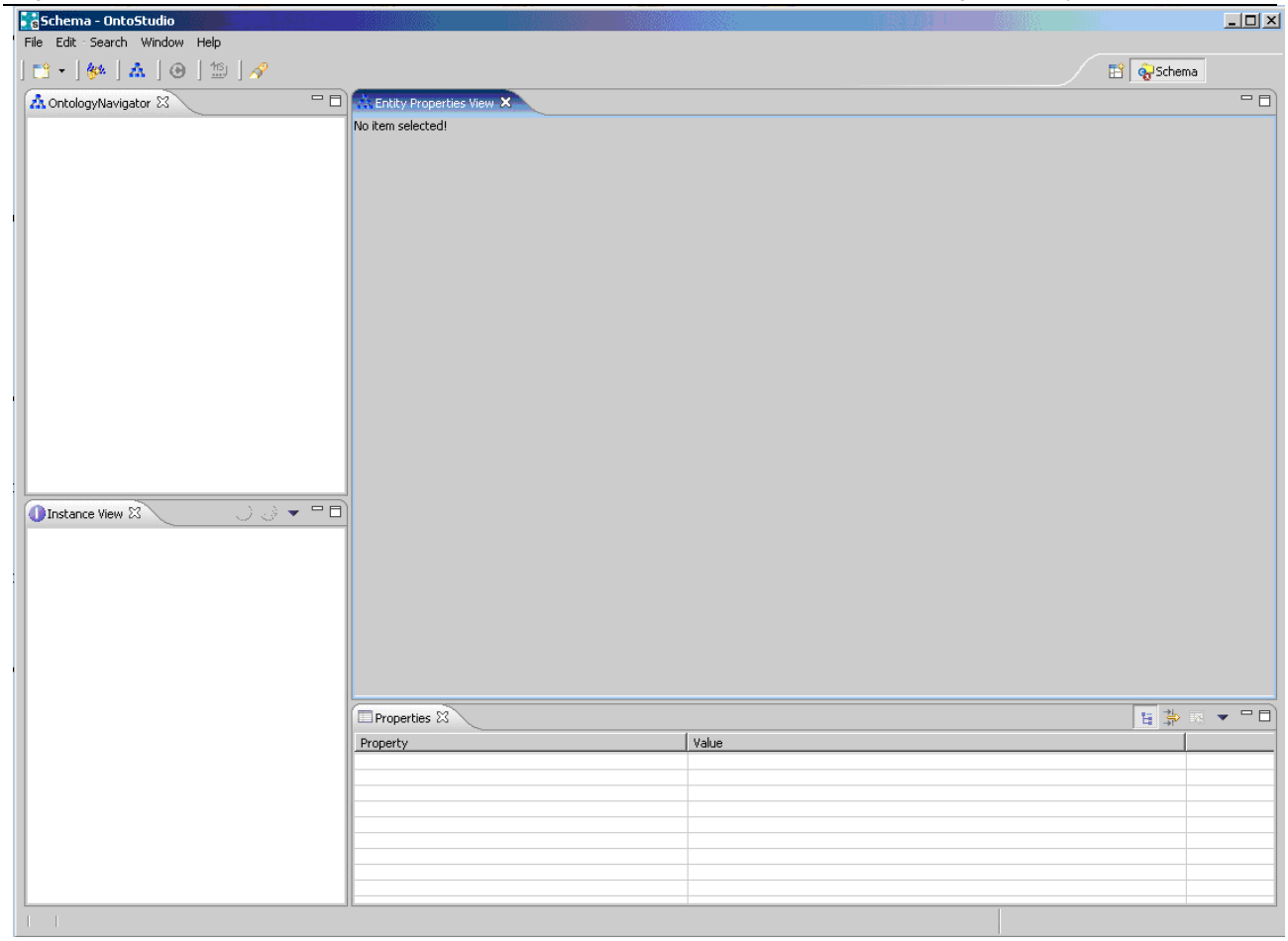
At the first start of the program the „Workspace Launcher“ window is loaded. There you can specify your workspace directory:


Workspace Launcher
Select a workspace
OntoStudio stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.
Workspace:
C:\Dokumente und Einstellungen\User\workspace
Browse...
Use this as the default and do not ask again
OK
Cancel

The selection of the workspace directory

In this dialog window you can select the directory in which NeOn Toolkit will save all its appropriate data. In order to save the data in a directory other than the specified one, click on the ***Browse…*** button and select the desired directory. Confirm with ***OK*** to start the application. The program starts and the start screen appears:

2007 © Copyright lies with the respective authors and their institutions.



Start screen of NeOn Toolkit – Schema Perspective

2.2. Hot Keys

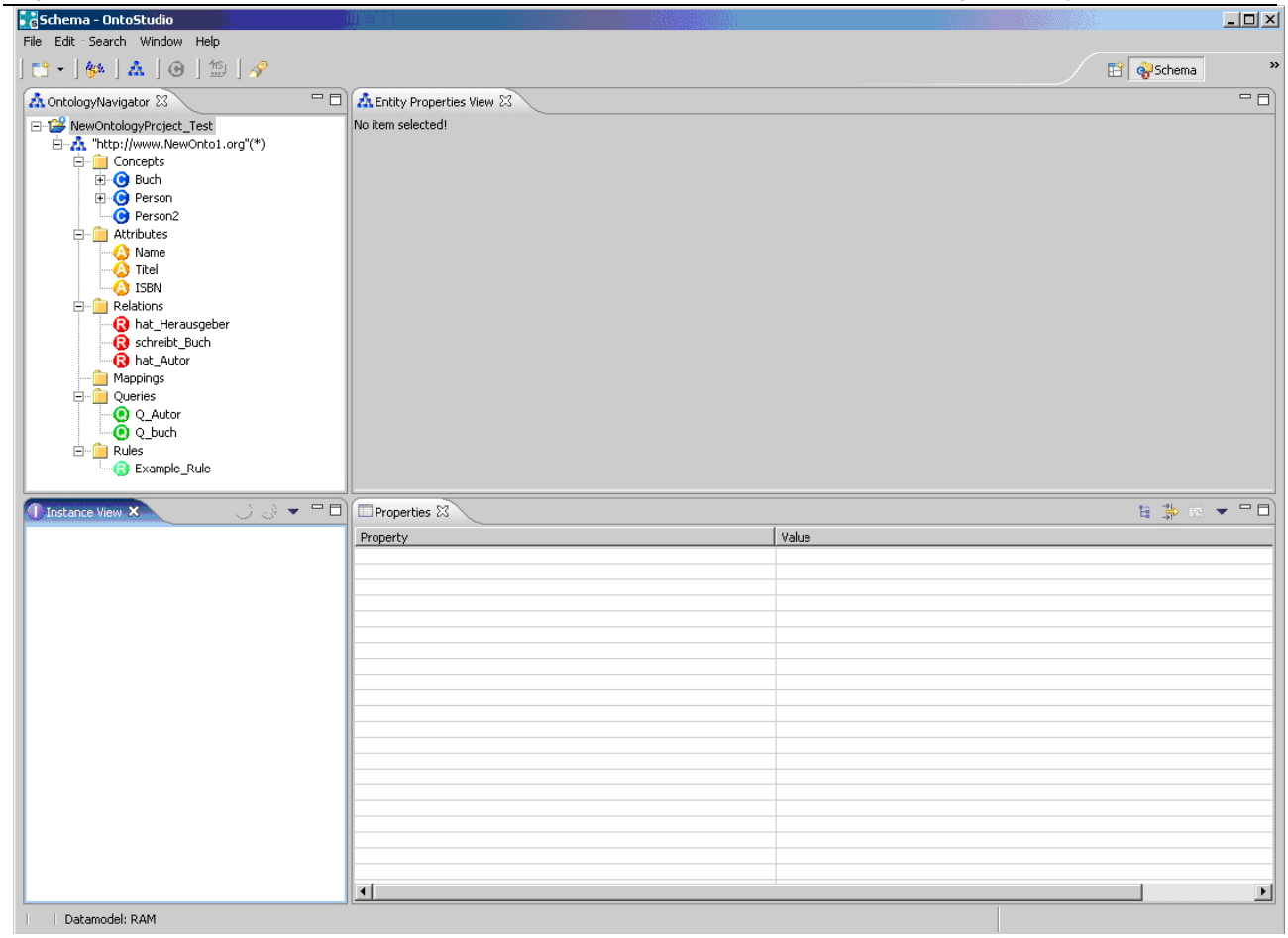
Apart from the usual hot keys, there are additional key combinations that allow for rapid and comfortable changing between the different views and editors, in order to simplify the navigation in NeOn Toolkit. They are listed in the following table.

Hot key	Function
File: Hot keys to create new data, projects, ontologies, etc.	
Alt + Shift + N	Menu: Create a new project
Ctrl + N	Menu: File - New
F5	Refresh: to actualize the view
Edit: Hot keys to edit the components	
Ctrl + X	Cut the marked text
Ctrl + C	Copy the marked text
Ctrl + V	Insert the text which is stored in the buffer
Del / Delete	Delete the marked text
Search: Hot keys to call the search function	
Ctrl + H	Open the search window
Window: Hot keys to change between the different menus	
ALT + -	Show the System Menu for the active view with the following activities: Fast View, Restore, Move, Size, Minimize, Maximize, Close
Ctrl + F10	Show the current view menu
Ctrl + M	Maximize the active view or the active editor
Ctrl + F7	Change to the next view
Ctrl + Shift + F7	Change to the previous view
Ctrl + F8	Change to the next perspective
Ctrl + Shift + F8	Change to the previous perspective
Ctrl + Shift + L	List of all implemented Eclipse short cuts

2.3. The User Interface

The user interface consists of a menu, a tool bar and the main area containing different so-called views for the actual content.

2007 © Copyright lies with the respective authors and their institutions.



The user interface of NeOn Toolkit (here: „Schema Perspective")

1. Menu bar and tool bar

   While the menu bar controls most operations of NeOn Toolkit, the tool bar contains frequently used operations. For example, you can open the „Mapping View" with one click on the appropriate button.

2. Main area

   As default, NeOn Toolkit starts with the „Schema Perspective" containing four different views: the „Ontology Navigator", the „Entity Properties View", the „Instance View" and the „Properties View".

3. Status bar

   The status bar at the bottom of the screen shows the current data model (usually set to RAM). Other datamodels that store models in a persistent data bases instead of in main memory, are available.

2.4. Perspectives

The features of NeOn Toolkit are grouped in different so-called *perspectives*, such as the „Schema Perspective" or the „Mapping Perspective". A perspective is a composition of different views to achieve a common task, e.g. the schema perspective contains views/tools for modeling ontologies.

On the right hand side of the toolbar you find the perspective selector which lets you switch between perspectives.

Button to display and change the main perspectives

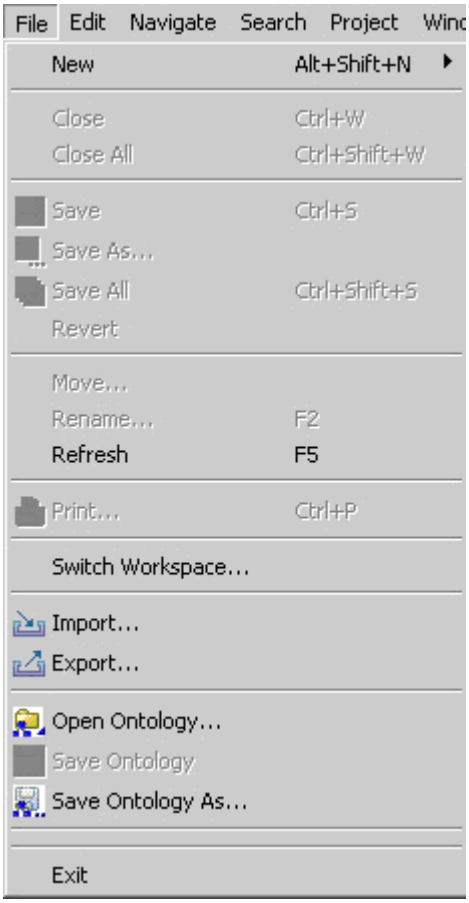
Currently NeOn Toolkit includes a couple of perspectives:

- The *Schema Perspective* displays general views for creating, editing and deleting ontology objects such as concepts, attributes, relations, instances, rules, and queries. When starting NeOn Toolkit, the „Schema Perspective“ is displayed by default.
- The *Mapping Perspective* to map between different ontologies or certain parts of them.
- The *F-Logic Editor Perspective* to edit a complete ontology by text in F-Logic.
- The *Debugging Perspective* to run queries in a debug mode and give the user the possibility of tracing.

2.5. NeOn Toolkit Menus

2.5.1.The „File“ Menu

The „File“ menu contains the standard operations of the program that are described below.

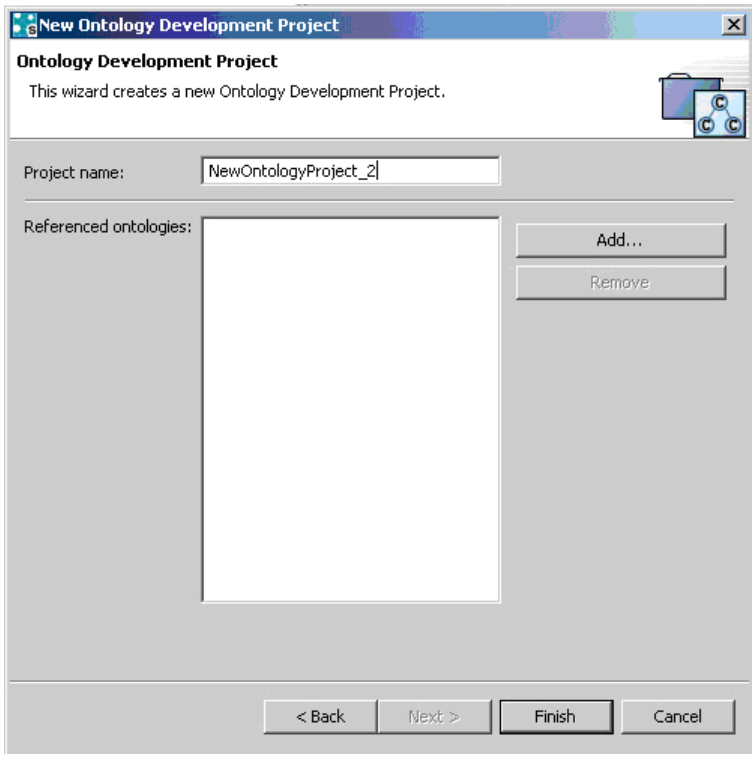


The „File“ menu

- File > New > Project > Ontology Project

2007 © Copyright lies with the respective authors and their institutions.

Creates a new ontology project: You must specify a project name that does not exist yet. If you want to add ontologies that already exist in other loaded projects, please click the ***ADD…*** button. Select the desired ontology in the following dialog. In order to remove an added ontology, mark this ontology and click the ***Remove*** button.



Creating a new ontology project

- File > New > Other

  Here you can create new elements, e. g., ontologies.

- ***Refresh***

  Updates the user interface, e.g. arranges the ontology elements alphabetically in the ontology navigator.

- ***Switch workspace***

  NeOn Toolkit stores your projects in a folder called a workspace. Choose a different workspace folder here to start a new session with the new workspace.

- ***Import Wizard***

  Opens the import dialog to choose a wizard for importing ontologies or other data sources

- ***Export Wizard***

  Opens the export dialog for writing ontologies in different formats

- ***Open Ontology***

  Opens an ontology from a selected file: This is the same functionality as ***File*** > ***Import*** > ***File System***

- ***Save Ontology as…***

  Saves the currently selected ontology: Represents the same functionality as ***File*** > ***Export*** > ***File System***

- ***Exit***

Before closing the application it displays a dialog where you can save the changes to your ontologies.

2.5.2.The „Edit“ Menu

This menu contains the usual functionality for cut, copy, and paste, as well as delete. These operations refer to the nodes in the ontology navigator.

2.5.3.The „Navigate“ Menu

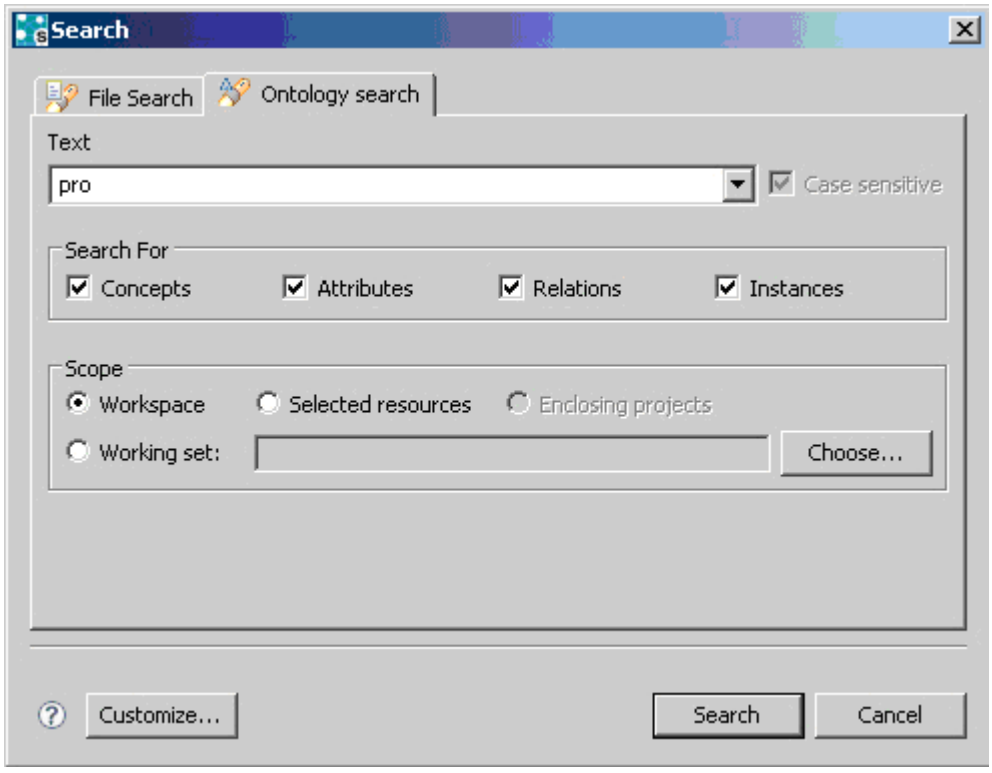
This menu entries of the „Navigate“ Menu are currently disabled.

2.5.4.The „Search“ Menu

With the „Search“ menu you can search for elements in ontologies.

The „Search“ menu

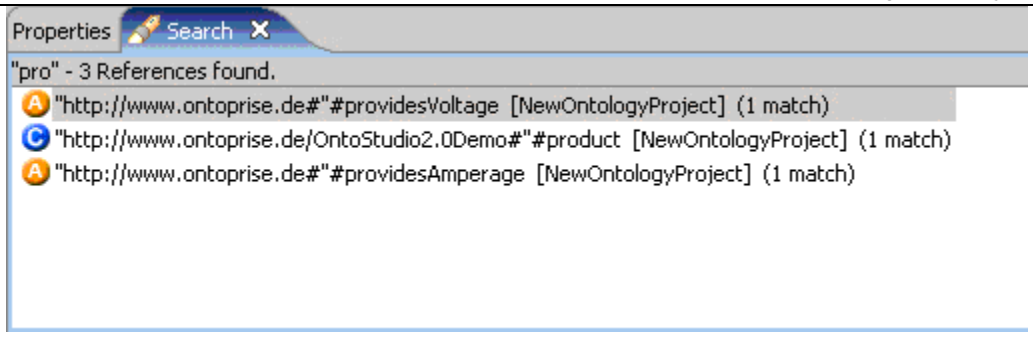
It opens the „Search“ dialog. This dialog contains tabs for searching in files and for searching in ontologies. The file-search option is not functional since we operate on the conceptual ontology model, rather than on files storing the ontologies.



The „Search“ dialog

You can search concepts, attributes, relations and instances inside the selected ontology. The results will be shown in a result view, where you can double click the entities to jump to the appropriate location in the ontology navigator..

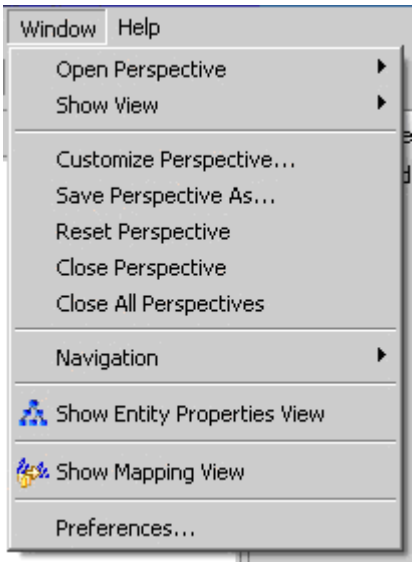
2007 © Copyright lies with the respective authors and their institutions.



The „Search - Result" dialog

2.5.5.The „Window" Menu

The Window Menu is a standard Eclipse feature to manage the visual components of the IDE. For a description of the menu entries refer to the Workbench user guide on the Eclipse web-site (cf. http://help.eclipse.org/help32/index.jsp).



The „Window" menu

- Open Perspective

  Opens the selection menu for perspectives
- ***Show View***

  Opens the selection menu for different views
- ***Customize Perspective***

  Opens the window to customize the different views
- ***Save Perspective As …***

  Opens the window to save the current perspective under the chosen name
- ***Reset Perspective***

  Deletes the preferences in the current perspective and restores the old preferences.
- ***Close Perspective***

  Closes the activated perspective

- ***Close All Perspectives***

  Closes all perspectives

- ***Navigation***

  Allows switching between different editor windows, views and perspectives

- ***Show Mapping View***

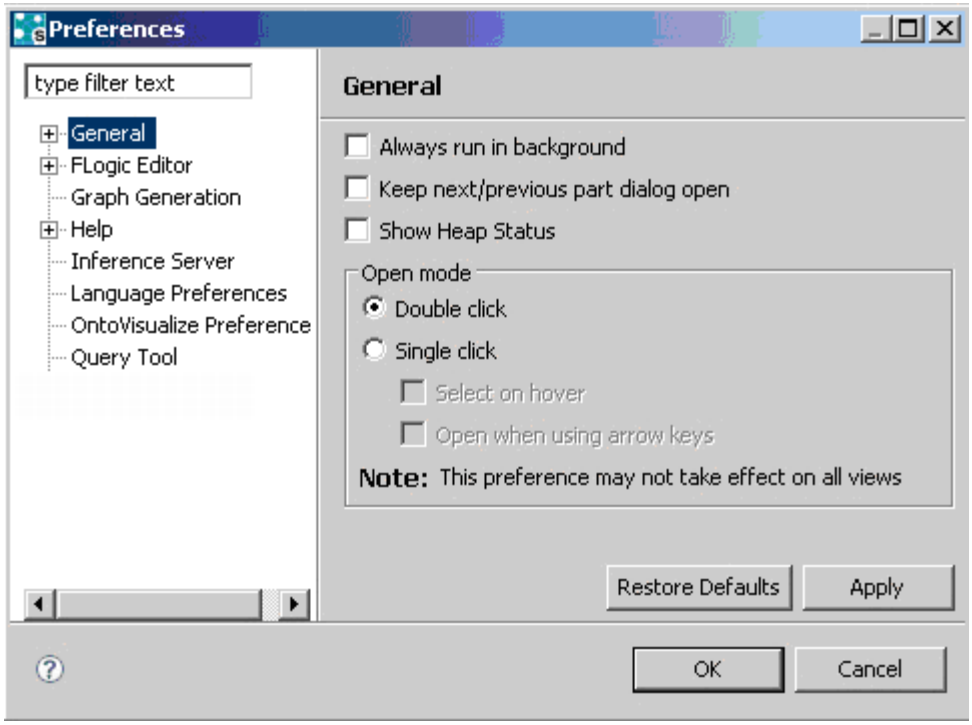
  The „Mapping View“ is displayed

- ***Show Entity Properties View***

  The „Entity Properties View“ is displayed

- ***Preferences***

  Opens the Eclipse standard preferences window. Some additional entries for NeOn Toolkit are available such as for the FLogic editor, language preferences, Ontology visualizer or the query tool. For a description of the other preferences refer to the Workbench user guide on the Eclipse web-site (cf. http://help.eclipse.org/help32/index.jsp).

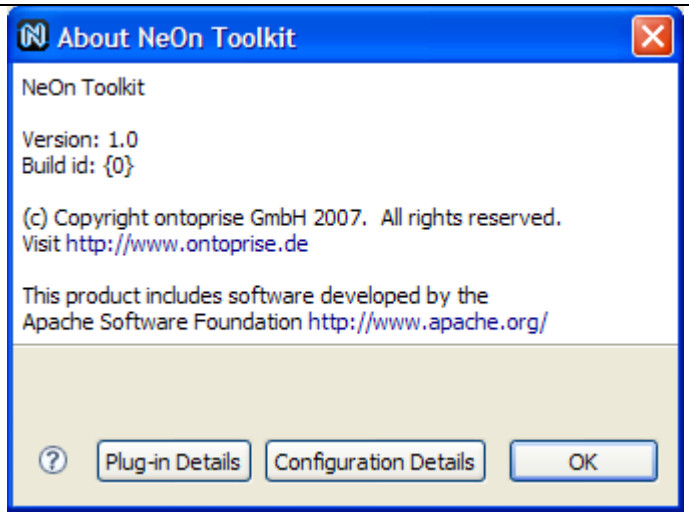


The „Preferences“ menu

2.5.6. The „Help“ Menu

Select ***Help*** > ***About*** to get further information about NeOn Toolkit. The following window shows the used NeOn Toolkit version with the build id, the contact data of ontoprise and the relevant copyright information. For further information follow the mentioned links.

2007 © Copyright lies with the respective authors and their institutions.



The „About" window

- Plug-in Details:

  A list of installed plug-ins and their versions

- Configuration Details:

  A list of configuration details of the installed NeOn Toolkit version. Here you can also open the error log file.

2.6. The Tool Bar of NeOn Toolkit

The symbols of the tool bar allow a quick access to some specific functions of NeOn Toolkit.

The symbol bar

- New

  The New button opens the „New" wizard where you can create a new ontology. It has the same function as the menu item ***File > New***

- Save

  disabled

- Print

  disabled

- Show Mapping View

  Opens the „Mapping View" window

- Show Entity Properties View

  Opens the „Entity Properties View" displaying details for the different elements of the ontology

- Generate Graph

  Generates graph starting at the given concept

- Create Concept

Create a new concept in the current ontology. If a concept is selected, a sub-concept is created. If the concept folder is selected, a new root concept is created.

A randomly generated name is proposed for the new concepts. In order to change the name directly, please enter a new name and then press Enter. The new concepts are not arranged in alphabetic order immediately. In order to sort them alphabetically, open the context menu for the „Concepts“ folder in the „Ontology Navigator“ and select the „Refresh“ menu item.

- Show/Hide Namespaces

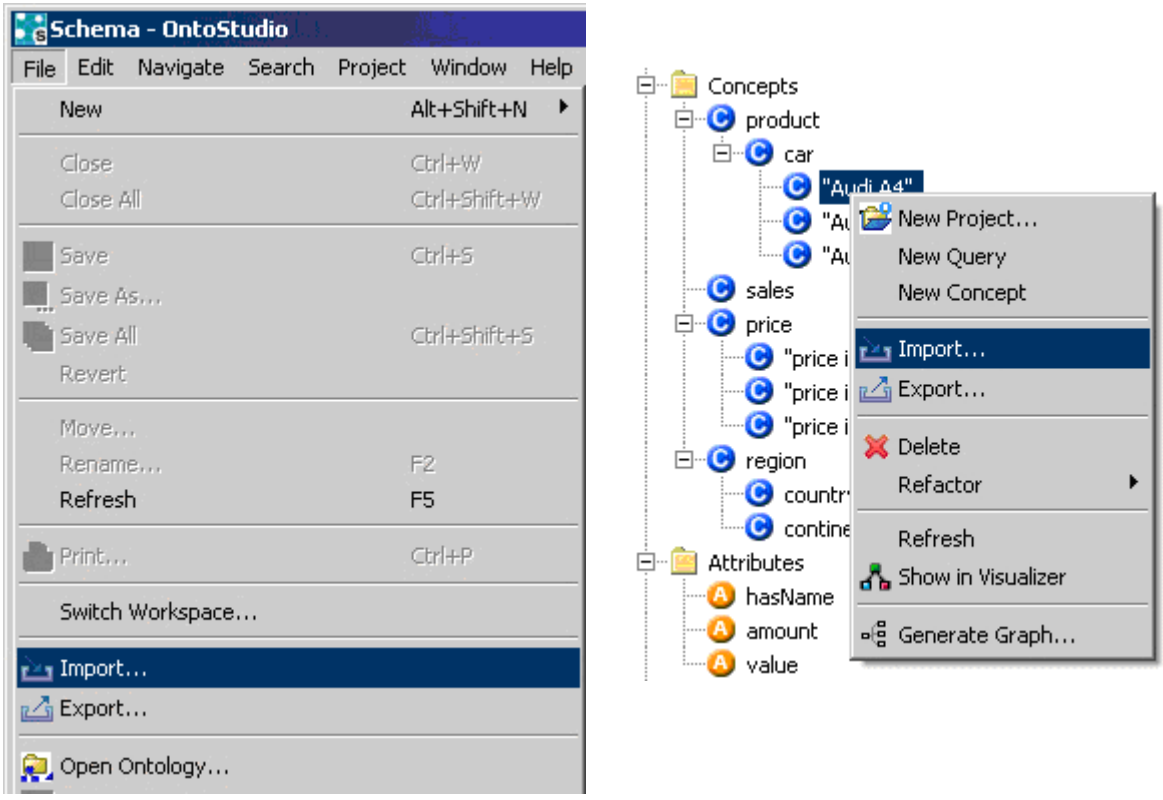
  Switches the display of the namespaces to „on“/“off“

- Search

  Opens the „Search“ window. Here you can search for elements in the selected ontology.

2.7. Importing Ontologies

Existing ontologies and other data structures such as database schemas can be loaded into NeOn Toolkit.

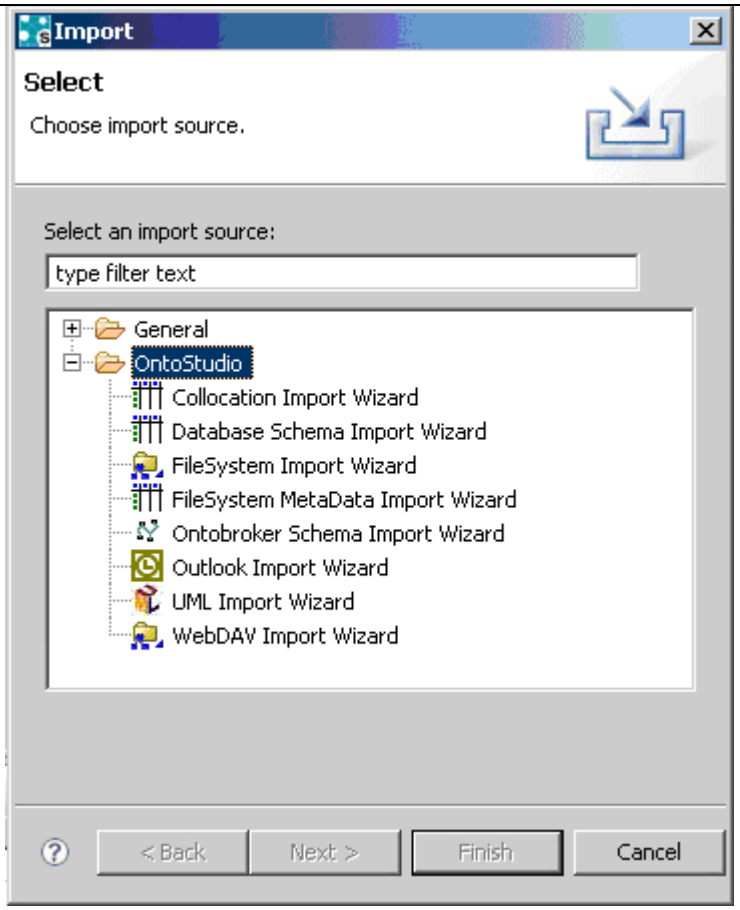


The menu window „File – Import...“ and the context menu of an element

You can initiate the import process following the following instructions:

- ***Select File > Import…*** on the menu bar or
- Select the entry ***Import…*** with a right mouse click in the „Ontology Navigator“ (the imported data source will be inserted into the project on which the import was applied)
- The „Import“ window opens to select different import wizards. Select the type of data for the import and confirm the appropriate procedure with ***Next.***

2007 © Copyright lies with the respective authors and their institutions.



The „File – Import…" View

2.7.1.Supported Import Formats

Currently we support

- Loading ontology files in different formats from the file system or a WebDAV server
- Translating a database schema from a relational database management system into an ontology
- Interpreting the folder structures and files from the file system as instances (FileSystem MetaData Import Wizard)

Some of these features are only supported by the extended configuration of NeOn Toolkit and not in the basic version.

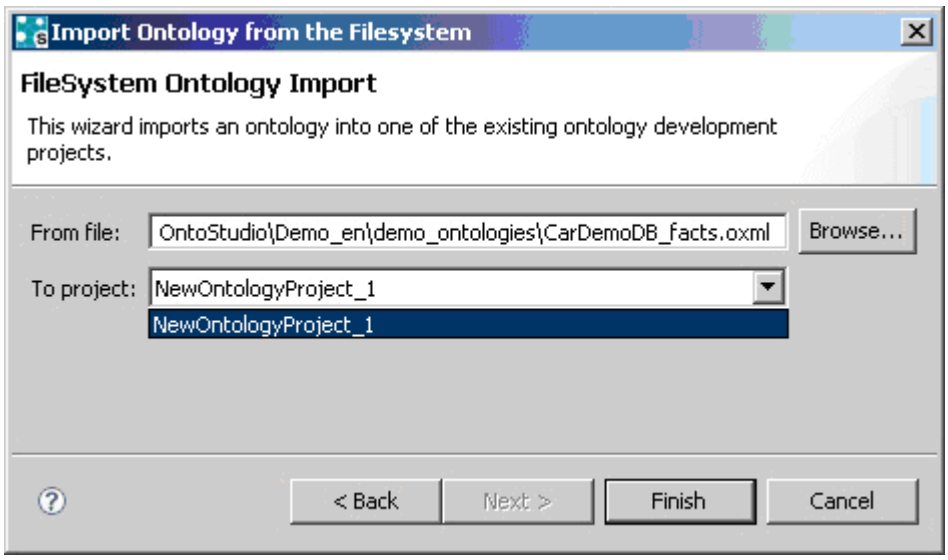
2.7.2. File System Import

For importing ontologies from the local file system, the following input formats are available:

- F-Logic – Frame logic is a knowledge representation- and ontology language (extension .flo)
- OXML – An XML-based ontology definition language used by ontoprise GmbH (extension .oxml)

- OWL – The web ontology language of the W3C (extension .owl)
- RDF / RDF (S) – Resource Description Framework (Schema) of the W3C (extension .rdf or .rdfs for the XML serialization or .nt for N-Triple or .n3 for N3)

To trigger this import, select ***File*** > ***Import*** > ***OntoStudio*** > ***FileSystem Import Wizard*** and click on ***Next.*** The subsequent dialog lets you choose the location of the ontology (***Browse*** button). You can also enter the location manually.



Dialog „FileSystem Ontology Import"

In the same dialog the drop down list „To project" refers to the project into which the ontology will be imported. After confirming with ***Finish*** the ontology will be imported. The selected ontology data source is imported from its location to the selected project and is now available within NeOn Toolkit.

2.8. Exporting Ontologies

The export function enables saving ontologies in different formats. For that purpose, you can use the corresponding menu items in the „File" menu.

You can export ontologies into the data formats F-Logic, OXML, OWL or RDF(s). If a project contains a number of ontologies (you can see this in the „Ontology Navigator" window), you must select the ontologies you wish to export by marking them specifically.

N.B.: Because of the differences of the ontology languages, information may be lost during the export process into the different formats, e.g. in the case of RDF-export rules which have been modeled are lost.

To initiate the export process from the „Ontology Navigator" select the ontology and open the context menu. Choose the entry ***Export…*** or choose this entry from the menu bar ***File*** > ***Export …*** In the following „Export" window select a destination to which the ontology will be exported (File System or WebDAV).

Export to FileSystem	It is possible to export to every directory which is available in the local file system.
Export to WebDAV Repository	A WebDAV Repository is a kind of remote file system that supports versioning and other sophisticated features. Thus, ontologies can be stored under the same name several times. This creates different releases of the same file.

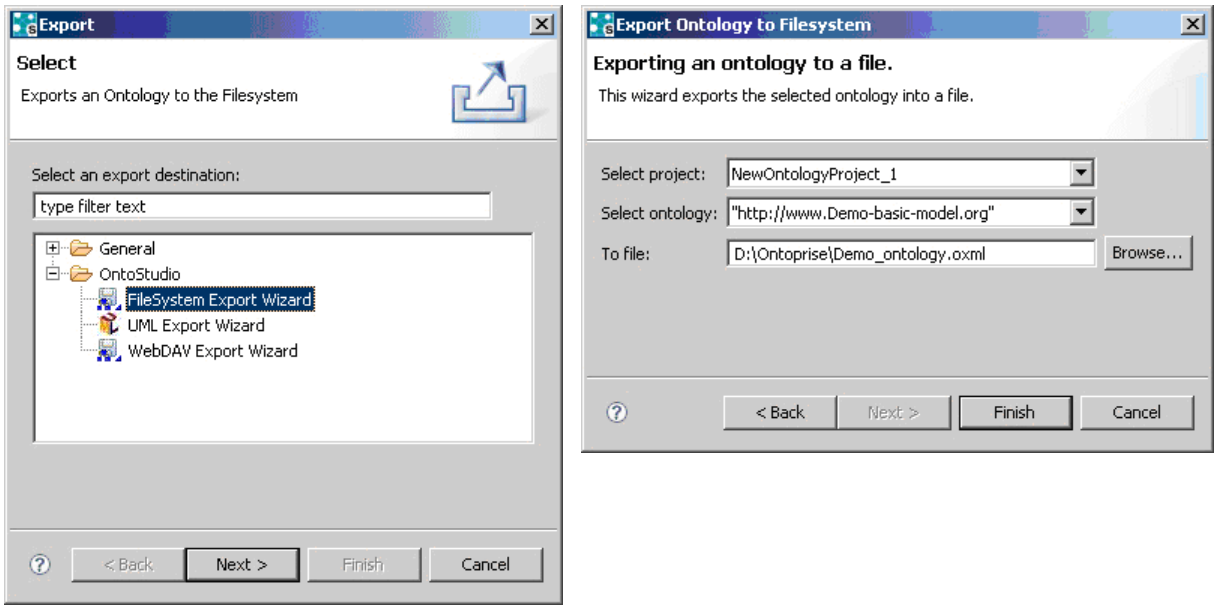
2007 © Copyright lies with the respective authors and their institutions.

2.8.1.Supported Export Filters

Currently we support the export of ontologies to the local file system or to a WebDAV repository using F-Logic, OXML, OWL or RDF(s). Also ontologies can be partially saved as UML class diagrams.

2.8.2.File System Export

For exporting ontologies to the local file system choose the File System Export Wizard in the Export menu and confirm with ***Next.***

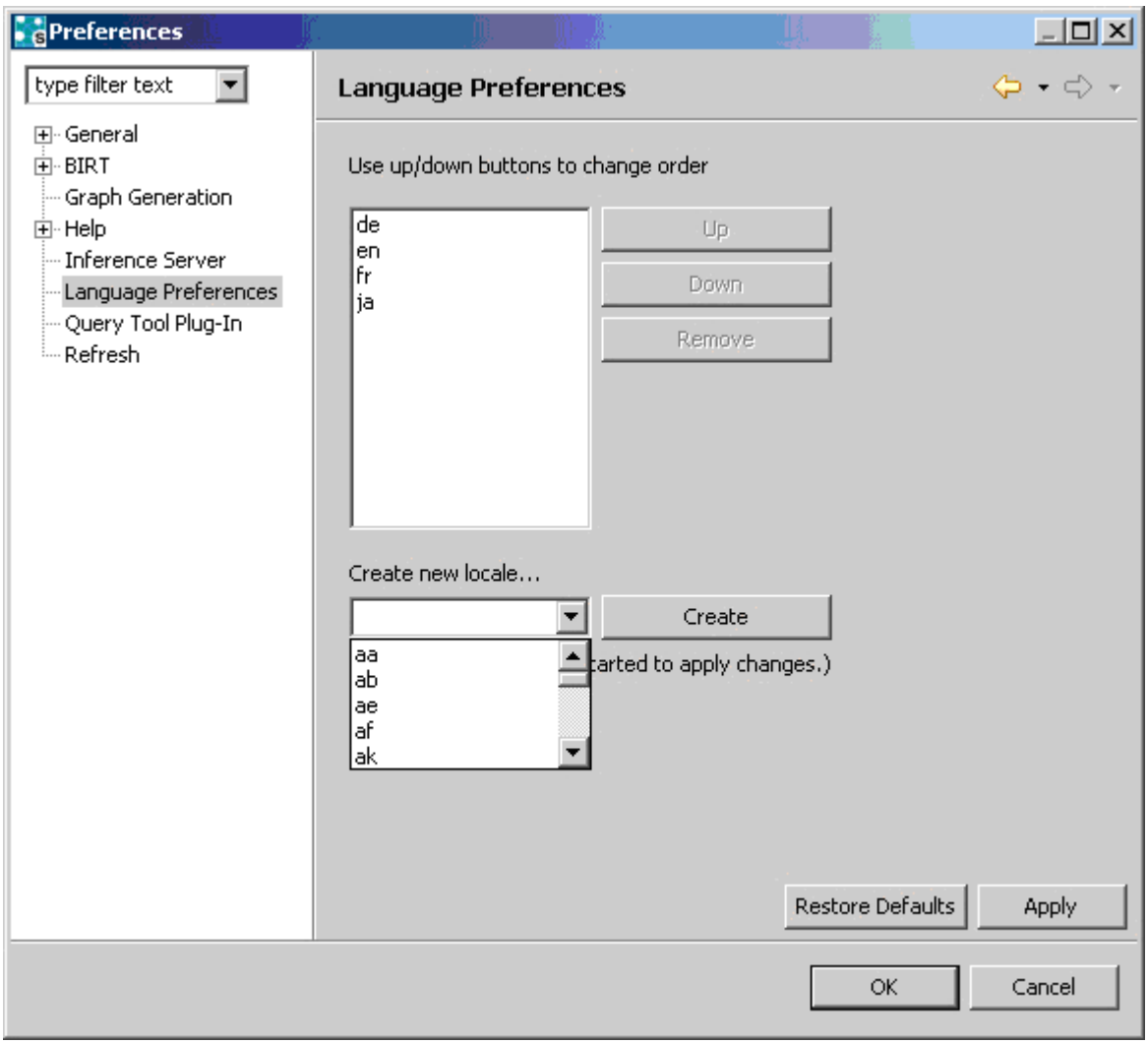


Dialog „Export File System"

In the drop-down lists „Select project" and „Select ontology" choose the ontology to be exported. Click ***Browse*** to select a target file with a desired data type („*.oxml", „*.flo", „*.rdfs", „*.owl") and confirm with ***Finish***. NeOn Toolkit will write the file to disk. Depending on the size of the ontology and the chosen format, the export process can take some time.

2.9. Localization of Ontologies

Within the language preferences you can specify the languages you want to develop your ontology for. To open the language preferences go to the ***Window*** menu, and select ***Preferences > Language Preferences.*** The fields for documentation and representation then let you input values for these languages. All languages according to ISO standard are supported!

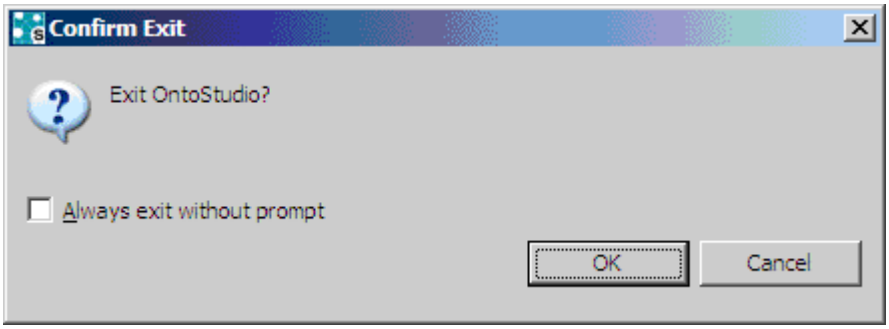


The „Preferences" menu to create or change the language

Mark the entries and push the ***Up*** or ***Down*** button to change the order. Create a new locale by clicking on the ***Create*** button.

2.10.How to exit NeOn Toolkit

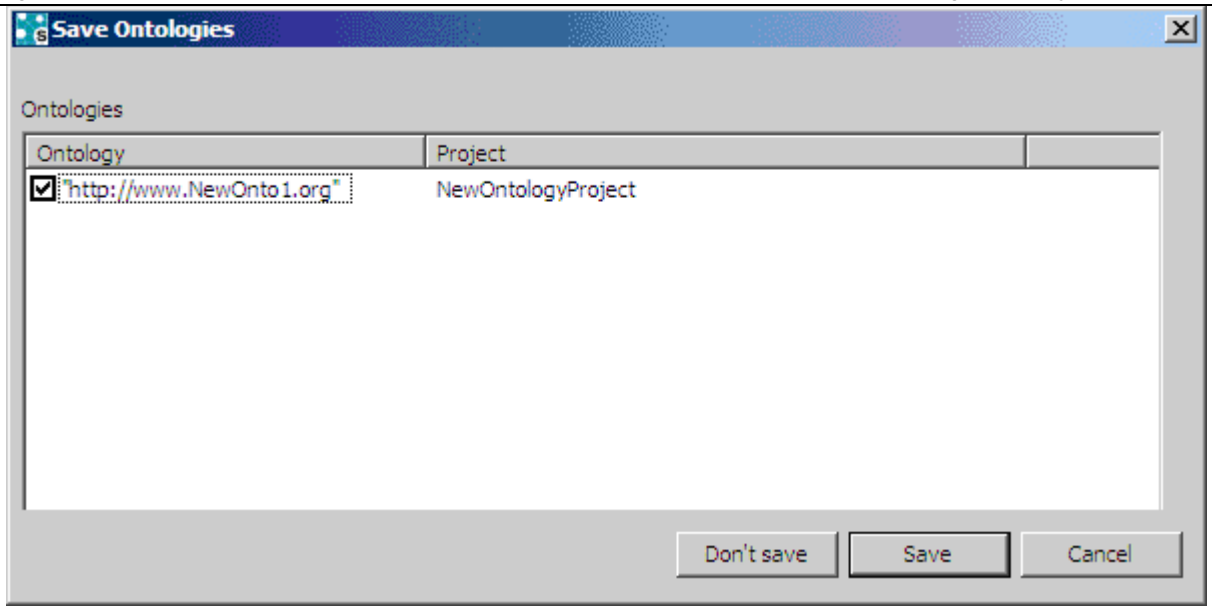
If you want to exit NeOn Toolkit, select ***File*** > ***Exit…*** in the menu bar and confirm with ***OK***. The „Confirm Exit" dialogue appears. Here you can set the checkmark „Always exit without prompt", in order to terminate NeOn Toolkit without being asked the next time.



The „Confirm Exit" window

If you created a new ontology, you will be asked for where you want to store it. Click on the ***Don't Save***, ***Save*** or ***Cancel*** button.

2007 © Copyright lies with the respective authors and their institutions.



The menu „Save Ontologies"

The application will be terminated.

3. Ontology Modelling

Creating a domain model requires creating a basic ontology, integrating existing information, eliciting knowledge from domain experts and formalizing it as rules. So the modelling process can be divided into four phases:

- the analysis of the domain
- the construction of the ontology
- the integration of already existing facts in databases
- the modelling of the expert knowledge as rules

The initial step in the modelling process is the analysis of the domain. It is necessary to exactly define the domain, which is a challenging task because the domain experts are in most cases not able to develop an ontology and the knowledge engineers do not have sufficient knowledge of the domain. To complete this task a knowledge transfer has to be done from the domain experts to the knowledge engineers.

The resulting system is based on an ontology. This ontology has two major objectives. Firstly it represents the terminology and the complex dependencies between the different domain parts. These dependencies are represented as relationships and rules. Secondly the ontology serves as a mediator between data from different sources for example to integrate up-to-date data about special domain parts etc.

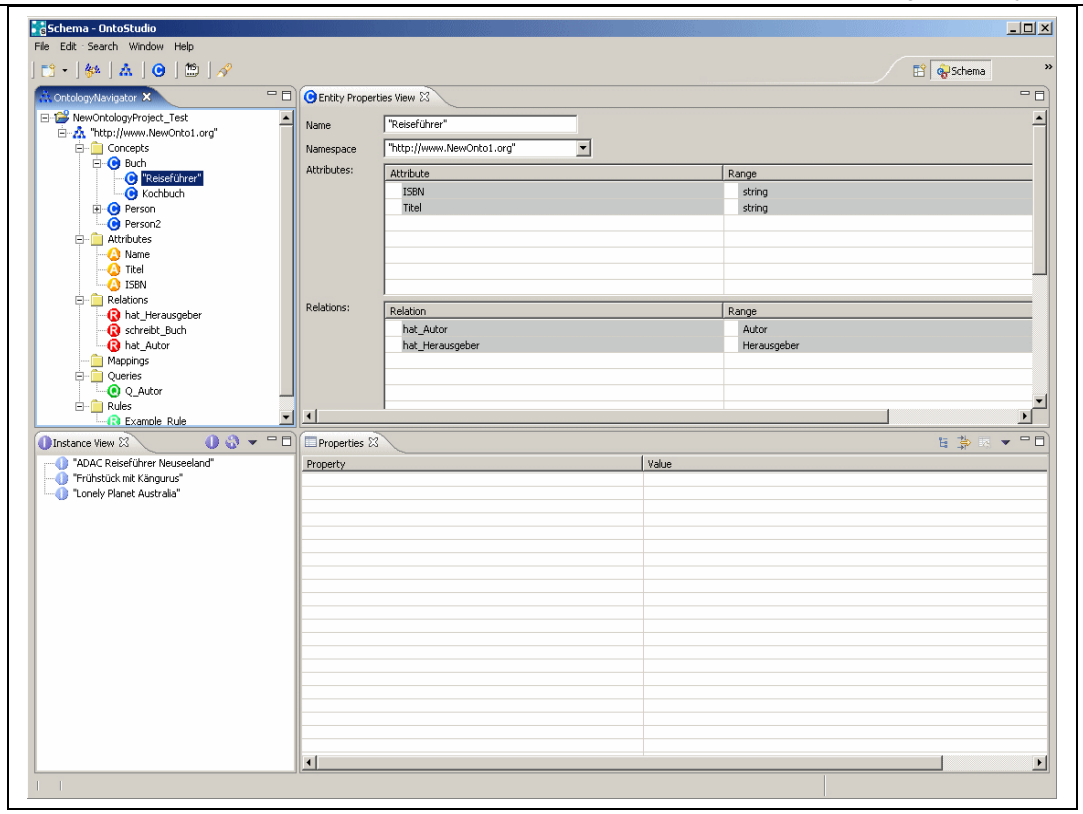
After the identification of the terms and properties of the domain the modelling phase starts. From here on NeOn Toolkit supports the whole engineering process, from the creation of the ontology to the development of the expert rules. At first you will create domain concepts and arrange them in a subsumption hierarchy. Afterwards attributes and relations will be added to the model and finally it will be completed with rules. Of course, this is not a strict sequence; lots of iterations are normal. The basic means for creating ontological knowledge is the schema-perspective.

3.1. Schema Perspective

By default, the Schema Perspective is split into four separate areas („views“):

- „Ontology Navigator“ (the tree representation in the top left)
- „Entity Properties View“ (top right)
- „Instance View“ (the list in the bottom left)
- „Properties View“ (to table in the bottom right)

2007 © Copyright lies with the respective authors and their institutions.



The „Schema" Perspective

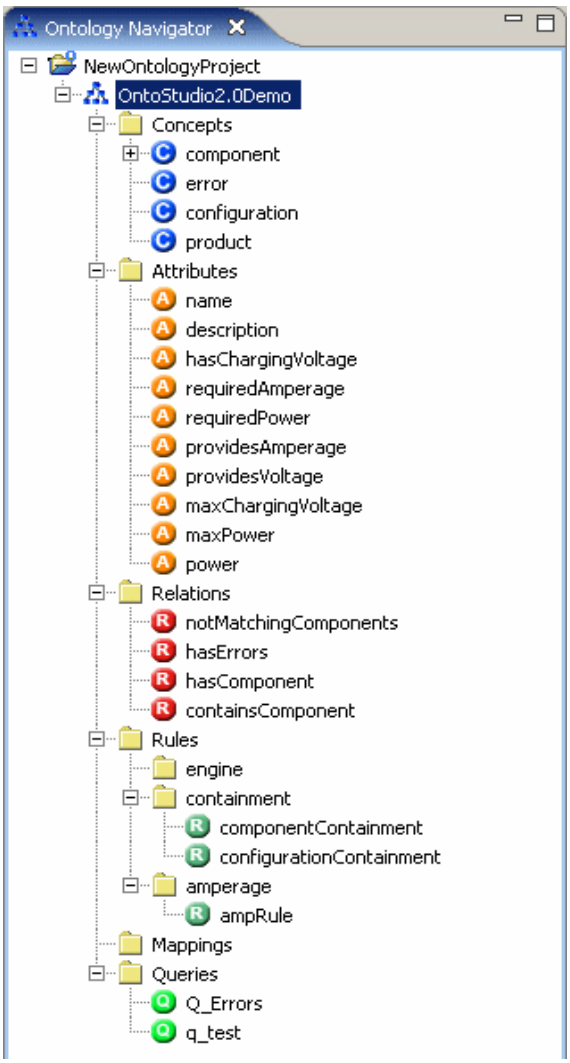
3.2. The Ontology Navigator

In the „Ontology Navigator" the different ontology projects and their corresponding ontologies are displayed. You can navigate within the hierarchically arranged elements as with the „Windows Explorer". Further you can create, change and delete the ontology elements. The different possibilities to edit the elements are listed in the appropriate context menu (To open the context menu of an element, click on the element with right mouse button).

If an element contains further sub elements, then a plus sign is displayed on the left side of the element. Please click on this plus sign to display the sub elements. In order to close the view of the sub elements, click on the minus sign of the parent element.

Rules are stored in the „Rules" folder. It is also possible to create your own rule subfolders with the adequate context menu to organize the rules.

Queries can be found in the „Queries" folder, and mappings in the „Mappings" folder.



The „Ontology Navigator" view of an ontology

For the different elements in the ontology navigator property views exist that present details and allow changing them. There are different „Entity Properties Views" for

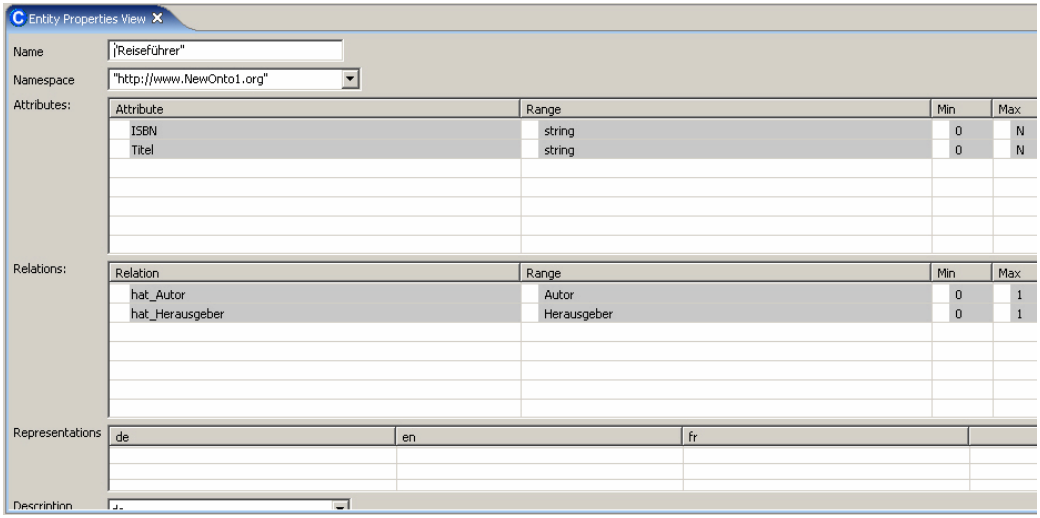
- concepts
- relations
- attributes
- rules
- queries
- mappings

The entity types rules, queries, and mappings are only available in the extended configuration of NeOn Toolkit and not in the basic version.

3.3. Entity Properties View for Concepts

In the „Entity Properties View" for concepts you can edit all properties of a concept. The table below explains all the different areas of the panel.

2007 © Copyright lies with the respective authors and their institutions.



„Entity Properties View" for concepts

Property	**Description**
Name	Name of the concept. Can be changed here. All changes will also be displayed in the hierarchy of the „Schema View".
Namespace	*Namespace* (URI) of the concept.
Attributes	In this area you can create attributes for your concept. Enter the name of the attribute in the „Attribute" column. Enter the range of values (String, Double, Boolean, Integer) in the „Range" column. All entries are accepted by means of the *Enter* button. With „Min" and „Max" you can define the cardinality of the attribute. You can delete an attribute by the red cross at the end of the line. Grey shaded attributes are inherited and can only be changed in the super-concept. All attributes will be displayed in the attributes folder in the „Schema View" as well. Here you also can rename them.
Relations	Here you can define relations, which describe the dependencies between concepts. Enter the name of the relation in the „Relation" column. Here, „Range" defines the other concept you want to relate to the active concept. All entries are accepted by means of the *Enter* button. „Min" and „Max" define the cardinality. The red cross deletes the selected relation. Grey shaded relations are inherited and can only be changed in the super concept.
Description	Textual documentation of the concept can be written in the description field. Multiple languages are supported, by default, the

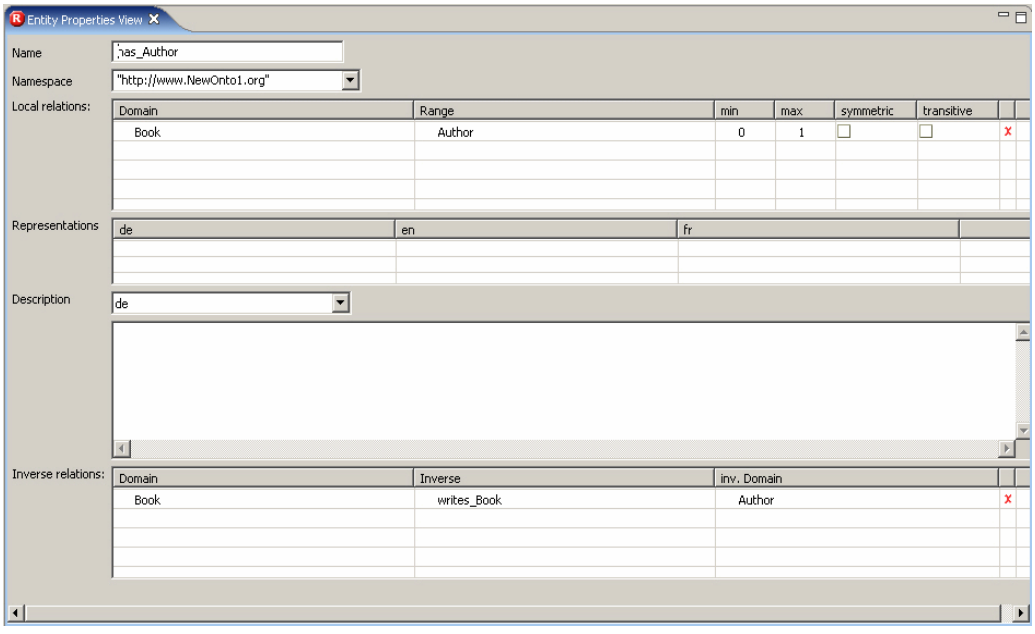
Property	Description
	documentation can be done in „English“, „German“ and „French“, but you can also add new languages by the NeOn Toolkit preferences.
Representations	A textual representation of the concept name in multiple languages.

The following table presents some examples for the different cardinalities of relations:

min	max	- Column
1	1	exactly one value (e.g. birth date)
0	1	none or one value (husband)
0	N	no value or optional number of values (children)
1	N	at least one value required or optional number of values (first name)

3.4. Entity Properties View for Relations

Relations are created within the context of a concept. But they can be edited starting from the relation folder of the Ontology Navigator. To open the „Entity Properties View“ for a relation, select the relation in the „Ontology Navigator“. This properties view displays information about the relations. All relations with the same name are displayed in *one* properties view.



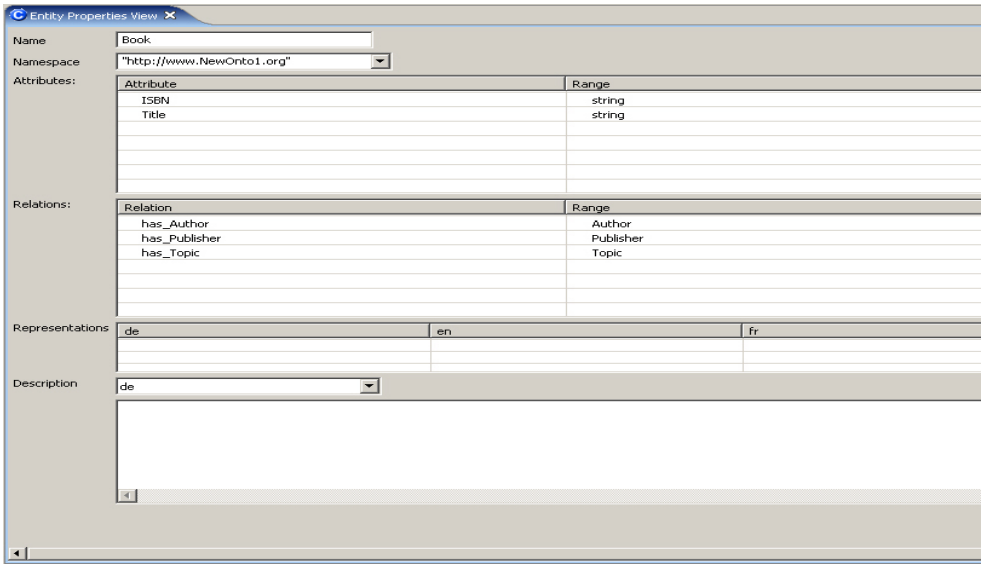
„Entity Properties View“ for relations

2007 © Copyright lies with the respective authors and their institutions.

Property	Description
Name	Name of the relation. Can be changed here. All changes will be also displayed in the hierarchy of the „Schema View“.
Namespace	*Namespace* (URI) of the relation.
Local relations	„Domain“ and „Range“ define the concepts connected by the relation. You can enter new concepts in the „Domain“ column. „Min“ and „Max“ define the cardinalities of the relation. The columns „symmetric“ and „transitive“ define whether the relations are symmetric or transitive. The red cross deletes a local relation between the two concepts.
Description	Textual documentation of the relation can be written in the description field. Multiple languages are supported, by default, the documentation can be done in „English“, „German“ and „French“, but you can also add new languages by the NeOn Toolkit preferences.
Representations	A textual representation of the relation name in multiple languages.
Inverse relations	The „Inverse“ column lists the relations that are inverse to the active relation.

3.5. Entity Properties View for Attributes

Essentially the entities properties view for attributes behaves in exactly the same way as for relations. The main difference is the restriction to only built-in types for the range field of attributes.

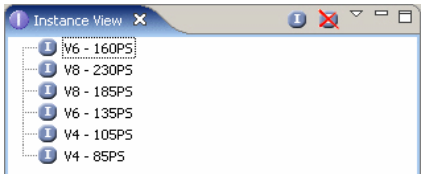


„Entity Properties View“ for attributes

Property	Description
Name	Name of the attribute. Can be changed here. All changes will also be displayed in the hierarchy of the „Schema View“.
Namespace	*Namespace* (URI) of the attribute
Local relations	The „Domain“ column displays the name of the concept for which the attribute is defined. New concepts can be added here as well. The type of the attribute (String, Double, Integer, Boolean) can be entered in the „Range“ column.
Description	Textual documentation of the attribute can be written in the description field. Multiple languages are supported, by default, the documentation can be done in „English“, „German“ and „French“, but you can also add new languages by the NeOn Toolkit preferences.
Representations	A textual representation of the attribute name in multiple languages.

3.6. Instance View

All existing instances for a concept will be shown in the instance view. New instances can be created, remove and deleted as well.



„Instance View“

3.6.1.Defining a New Instance

- Select the concept for which you want to create an instance in the „Ontology Navigator“ window by clicking on the concept.
- Activate the context menu by a right mouse click in the „Instance View“ and select „New Instance“ from the options menu or
- click on the ▫ symbol in the title line of „Instance View“
- In the „Instance View“ window a new instance is displayed, with a randomly generated name, e.g.: „instance1106663037379“.

2007 © Copyright lies with the respective authors and their institutions.

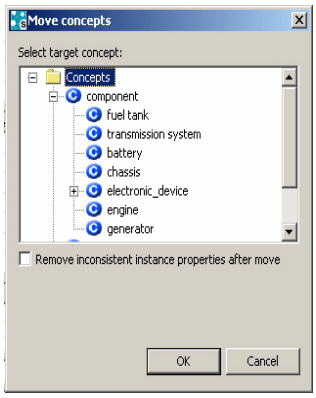
We recommend giving new instances an expressive name to be able to recognize them later in the ontology.

3.6.2.Deleting Instances

To delete an instance select it with a right mouse click and choose „Delete Instance“ from the context menu or mark the instance, which you want to delete and press the „Delete“ button in the tool bar of the „Instance View“.

3.6.3. Moving Instances

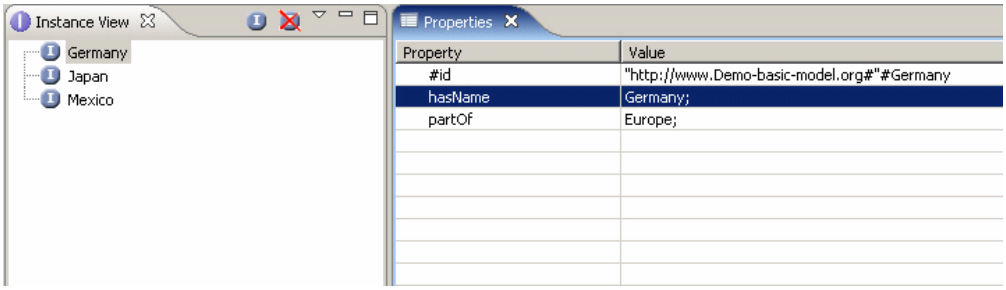
The instances of a concept can be moved by the „Move“ context menu entry or by „Drag&Drop“ from one concept to another. There you can choose whether to remove inconsistent properties of this instance as well as to remove inconsistent properties from other instances referring to the moved instances.



Menu – „Move instances“

3.6.4.Properties of Instances

You can change the properties of an instance in the „Properties View“.

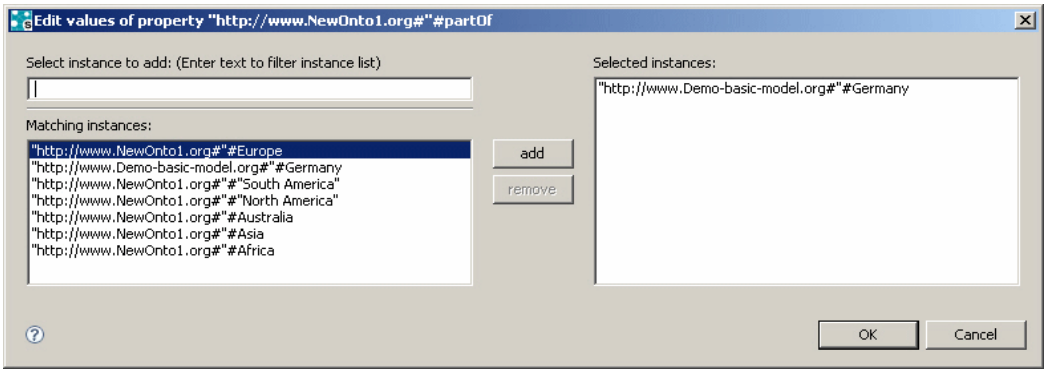


„Properties View“

If not already shown, you can open this view by ***Window*** > ***Show View*** > ***Other…*** > ***General*** > ***Properties*** from the window menu. The „Properties View“ displays two columns. The first shows available attributes and relations. The second displays the values of these attributes and relations and lets you modify, delete or add values.

Each instance has an „#id“-property, which is the unique identifier, consisting of a namespace and the name of the instance.

To change the value of one of the properties, click on its cell at the „Value“ column. If the selected property is an attribute/relation with a max cardinality of 1, you can enter the value directly into the cell. If its max cardinality is greater than 1, a multi-selection dialog opens, where you can enter/select multiple values.

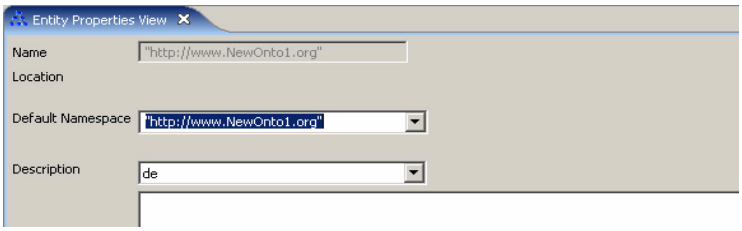


Multi select dialog

The „Matching Instances“ list shows the instances which are possible fillers for the relation. The list can be filtered by typing a letter into the „Select instances“ field, only instances starting with the given letter are still shown. To add an instance to the property, click on the matching instance and transfer it with ***add*** to the „Selected instances“ list.

3.7. Namespace Handling

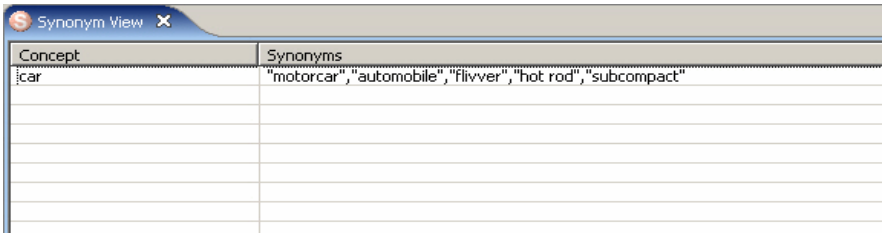
The default namespace of an ontology will be used when creating new ontology objects and can be changed for each object individually later. To change the default namespace for an ontology, you can change it in the entity property view of the ontology (after selecting the ontology in the ontology navigator).



„Entity Properties View“ of ontologies

3.8. Synonym View

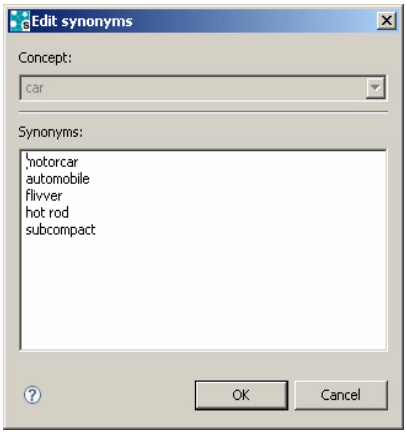
In the synonym editor you can add a list of synonyms for concepts, e.g. to reuse them in a search scenario (e.g. with „SemanticMiner“). To open the „Synonym View“ go to ***Window*** > ***Show View*** > ***Other...*** > ***OntoStudio*** > ***Synonyms***



2007 © Copyright lies with the respective authors and their institutions.

„Synonym View"

Select the concept you want to extend with synonyms in the „Ontology Navigator", the synonym view will update. By clicking on the synonyms cell in the „Synonym View", an editor appears where new synonyms can be entered or existing synonyms can be edited.

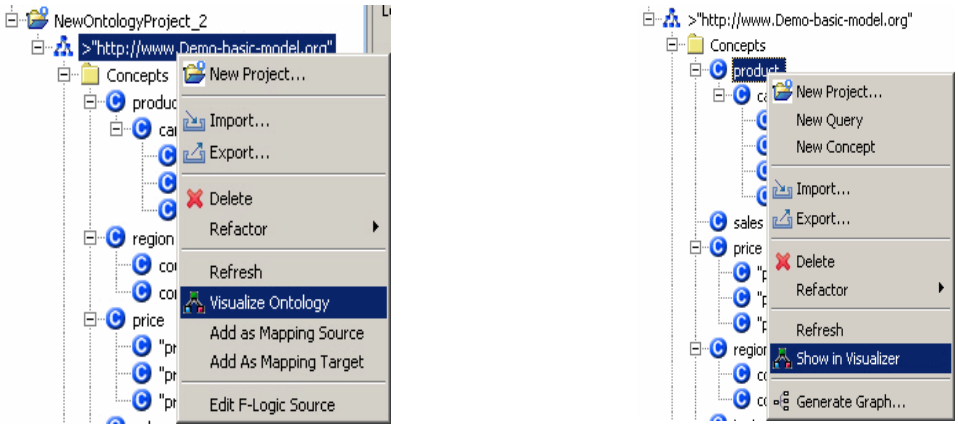


„Synonym Editor"

4. Visualizing Ontologies

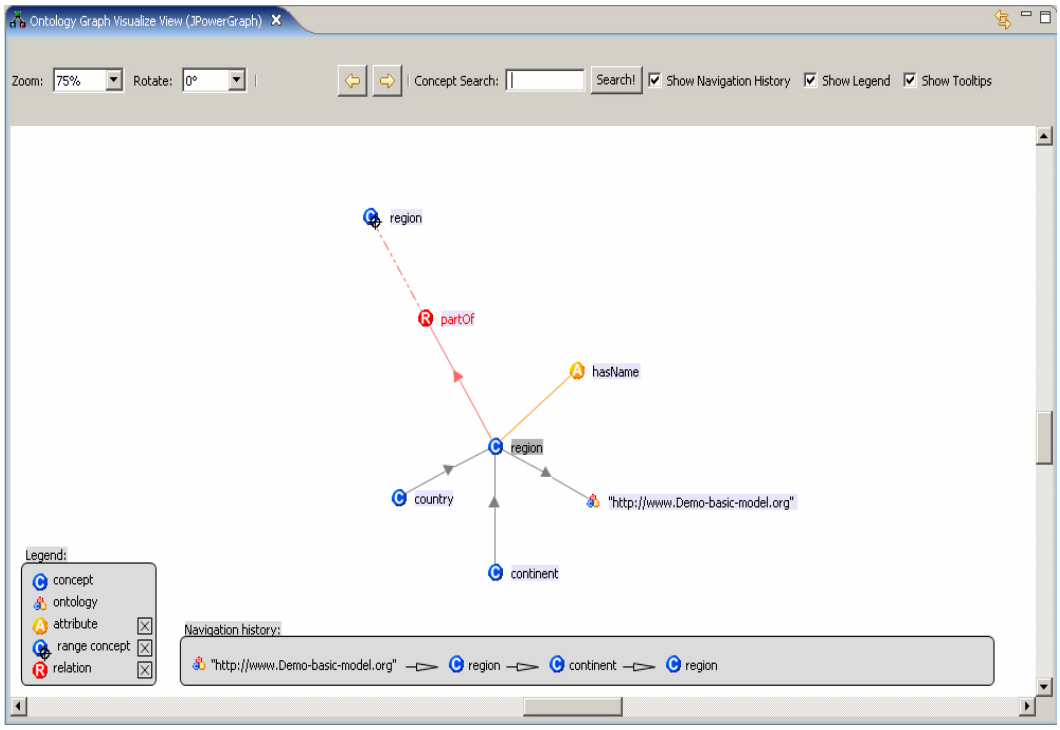
4.1. Ontology Graph Visualize View

The „Ontology Graph Visualize View“ displays the ontology and all elements of the hierarchy (concepts, relations, attributes, instances) in a graph. The values of attributes and relations of instances are not visualized. In the Ontology Navigator you can click with right mouse button on the ontology you want to visualize and select „Visualize Ontology“ or click with right mouse button on a single concept which you want to visualize and select „Show in Visualizer“.



Context menu to initiate the ontology graph visualizer

The „Ontology Graph Visualizer View“ will open, displaying the selected ontology or concept.

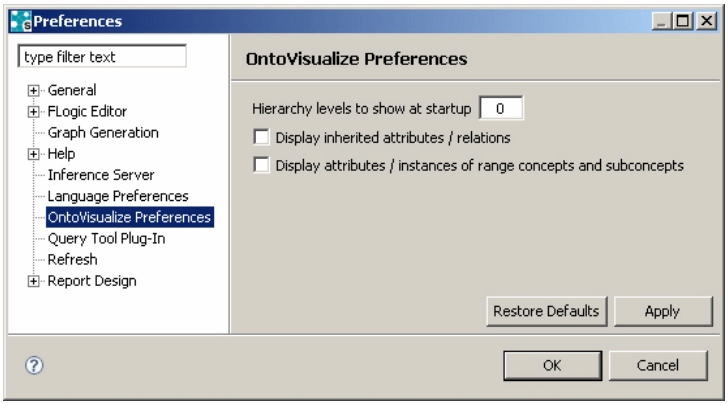


„Ontology Graph Visualizer“

2007 © Copyright lies with the respective authors and their institutions.

4.1.1.Concept in Focus

Clicking on a concept in the visualizer moves this element into the focus of the view and displays all adjacent elements, i.e. relations and attributes, as well as sub- and super-concepts. The depth of the sub-concept hierarchy displayed in the graph can be configured in the menu ***Window*** > ***Preferences…*** > ***OntoVisualize Preferences***.



„OntoVisualize Preferences"

4.1.2.Navigation history

At the bottom of the view a history of the last selected graph elements is displayed. By clicking on one of the elements here, the visualizer jumps back to the previous state. Stepping to the previous or next states is also possible by the arrow icons in the top pane.

4.1.3.Legend

At the left side of the view a legend is displayed explaining the icons used in the diagram. Here you can activate or deactivate different ontology entities by ticking the adequate checkboxes.

4.1.4.Graph elements

- **Concepts** are represented by blue circles, labeled with a C.
- **Attributes** are represented by orange circles labeled with an A.
- **Relations** are represented by red circles labeled with an R.
- The **ontology** is represented by a colorful icon consisting of tiny versions of the above three icons.
- **Instances** are represented by light-blue circles labeled with an I.
- Grey arrows represent the **concept hierarchy**, i.e. connections between super- and sub-concepts or root-level concepts and the ontology node. The arrowheads always point to super-concepts or ontology node.
- Orange edges represent the connection between concepts and **attributes**.
- Red arrows represent the connection between concepts and **relations**. A solid arrow points from the domain concept to the relation node and a dashed line refers to the range concept, which is additionally decorated with a little crosshairs icon.
- Light blue edges connect instance nodes with their concept nodes,

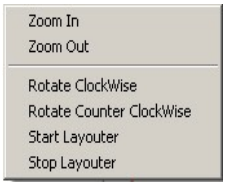
4.1.5.Other Functions

The visualizer also provides a **search feature**. After entering a concept name this concept is sought in the current diagram and it is put into focus. The search is restricted to concepts and is requires an exact match to successfully retrieve nodes.

In the preferences of the visualizer (***Window*** > ***Preferences…*** > ***OntoVisualize Preferences***) you can switch on two more features:

- Usually only direct attributes and relations of the concept in focus are displayed. Ticking the checkbox „Display inherited attributes/relations“ overwrites this and will also display properties that are inherited by this concept.
- The checkbox „Display attributes/instances of range concepts and subconcepts“ is responsible for adding also more detailed information for other concepts than the one in the focus.

The visualizer's context menu provides some additional functionality, like zooming in an out or rotating the whole graph.



Context menu of the „Ontology Graph Visualize View“

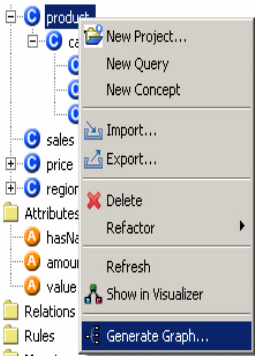
It also allows stopping the automatic layout algorithm and keeps all nodes at exactly the position where they are manually moved with the mouse.

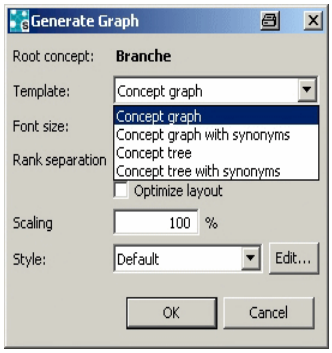
4.2. Graph View

The „Graph View“ provides printing functionality and allows exporting of a simple ontology graph as PDF document. Its main features are:

- Printing the concept graph
- Zooming in and out of the concept graph
- Displaying synonyms in the „Graph View“
- Searching for text in the „Graph View“

By clicking on the context menu of a concept in the „Ontology Navigator“, you can open the „***Generate Graph…“*** function. The selected concept and all its sub-concepts will be displayed in the „Graph View“.





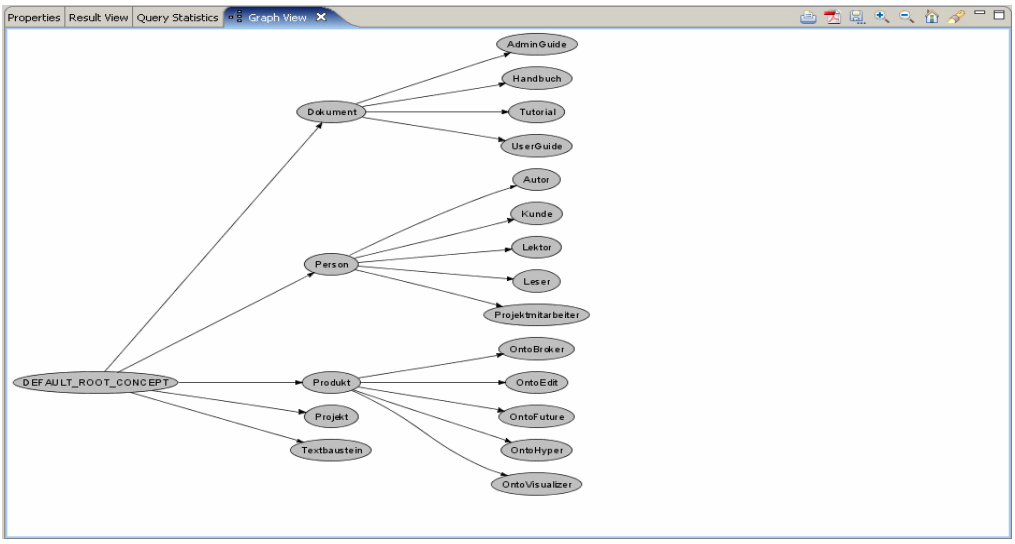
2007 © Copyright lies with the respective authors and their institutions.

Select „Graph View“ Settings for the „Graph View“

In the setting window you can select four different templates to visualize the graph diagram.

- Concept tree: Display of the hierarchic concept taxonomy
- Concept tree with synonyms: Display of the hierarchic concept taxonomy with synonyms
- Concept graph: Display the ontology with multiple inheritance
- Concept graph with synonyms: Display the ontology with multiple inheritance and the synonyms

The variants with multiple inheritance, will display concepts with multiple super-concepts only once and thus creating a graph structure. The other variants will duplicate concepts with more than one super-class and mark all but one occurrence as duplicates.



„Graph View with Template Concept Graph“

The icons in the top-right or the context menu entries let you print the graph, or save it as PDF or PNG. Also you can zoom in and out and search for concepts. In this view only hierarchical information is present, i.e. no relations, attributes or instances are available.

5. Rules

Domain models usually contain ontological knowledge and can be extended by adding rules with explanations, which represent complex knowledge going beyond the more structural properties of the ontology. Rules in general have a structure as follows:

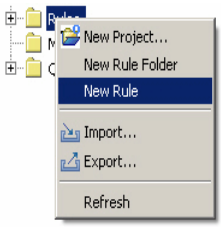
IF {the precondition is valid} THEN {the conclusion is valid as well}

The precondition is also called if-section or *body* of the rule. The conclusion is also called then-part or *head* of the rule. In order to specify a valid rule both parts are needed: body *and* head. Thus, it is necessary to specify which diagram elements are part of the rule body and which are part of the rule head. In the rule diagram, head elements are displayed in green, body elements are displayed in grey.

Rules are only supported by the extended configuration of NeOn Toolkit and are not available in the basic version.

5.1. Creating a Rule

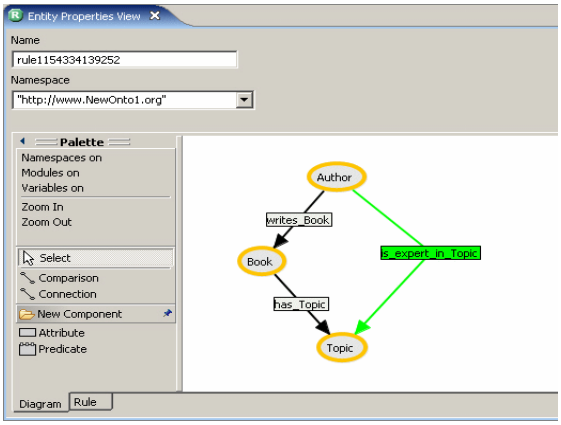
Right mouse click to the „Rules“ folder of the Ontology Navigator and select the „New Rule“ entry. You can change the proposed random name.



Context menu „New Rule“

5.2. Entity Property View for Rules

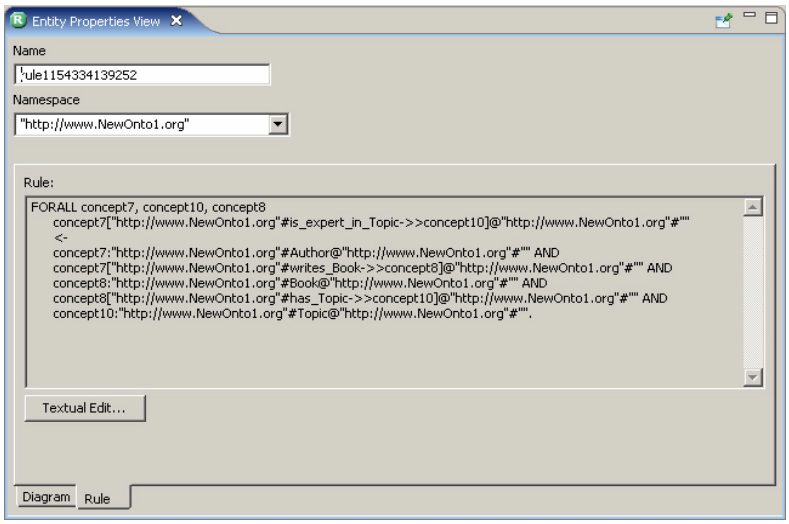
To switch to the Graphical Rule Editor (GRE), click on a rule in the „Ontology Navigator“. This action opens the „Entity Properties View“ where the GRE is displayed for the corresponding rule. The chosen rule will be displayed with name, namespace and the rule diagram.



2007 © Copyright lies with the respective authors and their institutions.

„Entity Property View" for Rules

Clicking on the „Rule" tab on the bottom will activate the textual rule editor. Clicking on the „Diagram" tab will activate the graphical editor again.

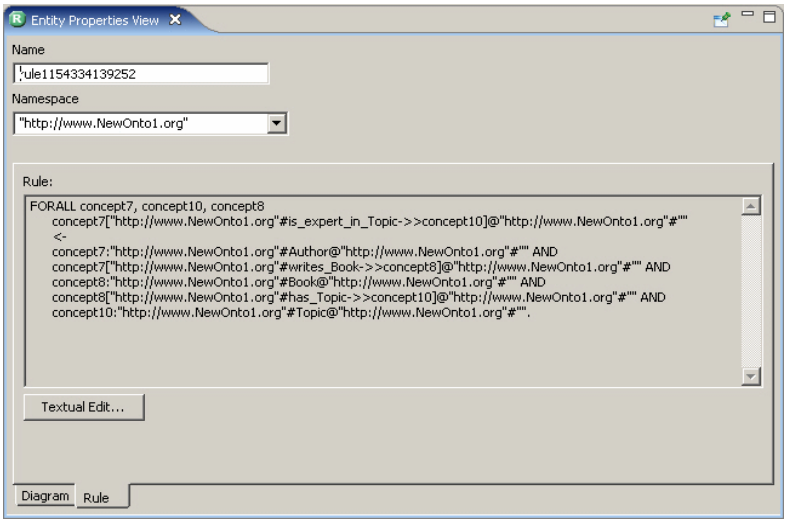


„Rule" Register Tab

Use the ***Fix*** button in the top right corner to make the GRE view "sticky", i.e. to keep this rule displayed, even if you select other elements from the ontology navigator. This is needed to drag and drop concepts or instances into the graphical editor. Simply unselect the Fix icon and the entity property view will automatically update according to the selected elements in the ontology navigator again.

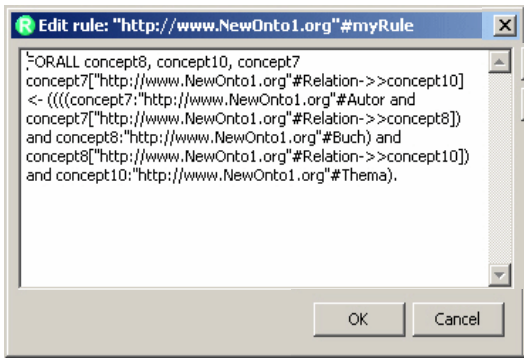
5.3. Textually Editing Rules

The Rule tab displays a textual representation of the rule in F-Logic.



Textual view of a rule in its „Entity Properties View „

Clicking on the button ***Textual Edit…*** opens an editable text field where the F-Logic representation of the rule can be modified.



„Textual Edit …" view

The Graphical Rule Editor does not support arbitrary F-Logic rules. This implies that some rules must be formulated textually and these rules cannot be displayed correctly with the graphical view. Rules making use of function symbols, or complex first order formulas in the rule body are examples for rules not supported by the graphical representation.

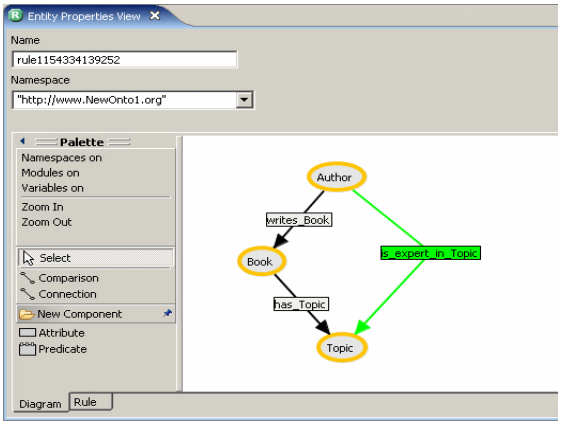
For a full description of the F-Logic Syntax and Semantics refer to the ontoprise web-site www.ontoprise.de.

5.4. Graphical Rule Editor

5.4.1.Overview

The main means for creating and modifying existing rules is the Graphical Rule Editor (GRE). It enables you to formulate rules in a graphical, interactive way. In the following we show how rules can be modeled with the help of the GRE. These example rules use the domain of books, their topics, publishers and authors to illustrate the GRE. The example provides an insight into the modeling process and should familiarize you with the major functions of the GRE.

On the left side of the GRE you find the palette with several text buttons that provide the functionality of the GRE.



2007 © Copyright lies with the respective authors and their institutions.

Graphical representation of a rule in its „Entity Properties View"

5.4.2. Adding Concepts and Instances to the GRE

To add a concept node or an instance to the rule diagram simply „Drag&Drop" it from the „Ontology Navigator" or from the „Instance View" into the GRE. The new node appears in the diagram where you dropped it.

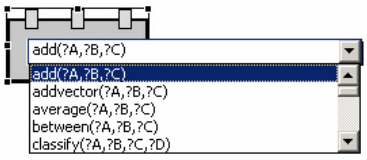
Make sure the diagram is sticky, i.e. the pin-icon in the top right corner of the rule entity property view is selected.

5.4.3.Adding Attributes to the GRE

To add attribute nodes to the diagram choose „Attribute value" from the palette menu and click into the diagram to create an empty attribute node. Fill in desired values and connect the attribute node with the corresponding concept by adding a connection and selecting the name of the desired attribute. Attribute nodes support the following data types: „boolean", „datetime", „time", „date", „number", „integer", and „string".

5.4.4.Adding Predicates to the GRE

For adding predicate nodes to the diagram choose the „Predicate" entry from the palette menu and then click into the diagram to create a preliminary predicate node. By clicking on the label of the node a drop down list appears, which lets you choose the desired built-in predicate with the appropriate number of arguments.



The selection list of a predicate

For every argument of the predicate a square appears on the top of the predicate. These *anchors* serve to connect the predicate with the rest of the rule diagram. By adding a connection between the anchor and other elements of the diagram, such as instances or attribute values, the arguments of the predicate are bound.

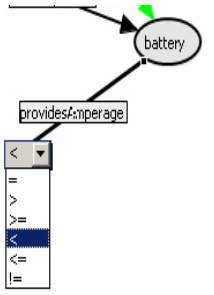
A complete list of the built-in predicates is presented in the OntoBroker User Guide.

5.4.5.Adding Relations or Attributes to the GRE

Two concept nodes can be connected by a connection. This connection represents a relation between instances of the respective concepts. To establish such a relation select the „Connection" element from the palette menu and then click on the source and the target concept. This introduces a new directed connection into the diagram. The label of the connection can be changed to represent the actual semantics of the relation. When selecting the label, a drop down list appears containing all valid relations. A relation is valid if the ontology defines the source

concept as the domain and the target concept as the range of this relation. Of course inheritance is considered, too, when computing the list.

Attribute edges are created in exactly the same way as relations. The only difference is that the target must be an attribute value node, not a concept node. Also, attribute connections contain comparators at the end where the value node is located. The comparators specify the semantics of the link. These comparators are available: =, <, >, !=, ≤, ≥.



Choosing the comparator of an attribute connection

If the comparator reads “<”, and the value reads 24 then the attribute edge of the diagram above reads “If the battery provides amperage less than 24…”.

5.4.6.Adding a Comparison

Comparison edges simply express equality or inequality of two nodes. They can be established via the “Comparison” button of the palette. They are created in the same way as attribute or relation links but their label does not come from the ontology but from the set {=, <, >, !=, ≤, ≥}. Comparison edges are only valid in the body of a rule and, thus, specify additional conditions. They can also be used to specify that two concept nodes actually represent the same instance.

5.4.7.Specifying Head and Body Parts

A rule always consists of conditions and conclusions. The precondition is also called *if-section* or *body* of the rule. The conclusion is also called *then-part* or *head* or the rule. In order to specify a valid rule both parts are needed: body *and* head. Thus, it is necessary to specify which diagram elements are part of the rule body and which are part of the rule head. In the rule diagram, head elements are displayed in green, body elements are displayed in grey.

Relation or Attribute edges as well as concept nodes can be moved from the body to the head and vice versa. To do so, select the element, open the context menu and choose ***Toggle If-/Then-section***. The element should change its color from grey to green or back.

Attribute-value nodes, predicated nodes and comparison arcs cannot be moved to the head of rules.

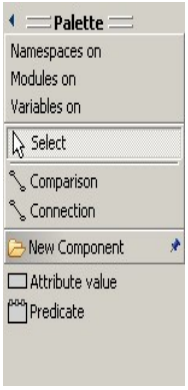
5.4.8.Negations

Sometimes it is necessary to express conditions that do **not** hold. This can be done by selecting a rule element (which must be in the body of the rule), opening the context menu and choosing ***Toggle negation***. Negated elements are represented in red.

5.4.9.The Palette Menu

The editing tools for the GRE are provided in the palette menu.

2007 © Copyright lies with the respective authors and their institutions.

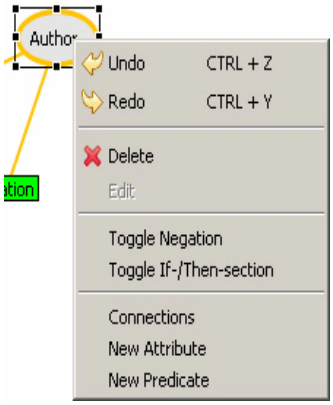


Menu „Palette"

- Namespaces on: Switches the presentation of the „Namespace-Prefixes" on and off. Namespaces are displayed for concept and property names, and also for instances nodes.
- ***Modules on***: Switches the presentation of the „Module-Prefixes" on and off. Each F-Logic-Statement refers to a module. Usually this information is not relevant, because only one module is used. In cases where, e.g. information from more than one module is integrated this module-information is very relevant. Module information is available for concept and predicate-nodes, as well as for attribute and relation arcs.
- ***Variables on***: Switches the presentation of „Variables „ on and off. If off, all variables are implicit only. If on, variables representing the instances of concepts (in concept nodes) are explicitly shown in the diagram.
- ***Select***: Activates the mouse selector.
- ***Comparison***: Activates the comparison selector to draw comparison arcs (=, <, >, !=, ≤, ≥) between two concepts nodes.
- ***Connection***: Activates the connection selector to draw attribute and relation arcs between concept nodes or between concept and attribute nodes.
- ***New Component***: Shows / hides the node types on the bottom of the palette.
- ***Attribute value***: Lets you create new, empty attribute value nodes in the rule graph.
- ***Predicate***: Lets you create new predicate nodes in the rule graph.

5.4.10.The Context Menu

With a right mouse-click on any element in the GRE diagram, a context menu opens, containing the following entries:



Workflow functions of the GRE desktop

- Undo: Undo the last operation
- ***Redo***: Redo the last operation
- ***Delete***: Remove the selected element(s) and possible incoming and outgoing arcs
- ***Edit***: Edit the selected element, only active for connections, attribute values, and predicates
- ***Toggle Negation***: Operation to negate the selected elements, i.e. concept nodes, attribute or relation arcs, or predicate nodes. Negated elements are displayed in red.
- ***Toggle If-/Then-section***: You can mark elements to be part of the rule body or head. Elements in the body (if-section) represent preconditions of the rule. Head-elements (in the then-section) define the results of a rule, i.e. statements that are derived by this rule. Body-elements are marked in grey. Rule elements in the head are represented in green.
- ***Connections***: Create a new connection arc.
- ***Attribute value***: Create a new attribute value node.
- ***New Predicate***: Create a new predicate node.

5.5. Stylized English View

The Stylized English View shows a natural-language-like representation of the rule diagram currently shown in the Graphical Rule Editor. It can be used to read aloud and verify the contents of the rule diagram. If the view is not visible go to ***Window*** > ***Show View*** > ***OntoStudio*** > ***Stylized English View*** to display it.

2007 © Copyright lies with the respective authors and their institutions.

6. Sample Walkthrough

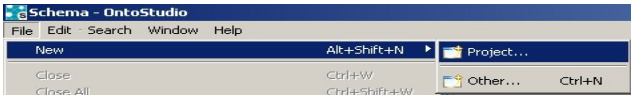
In this section we will build a small ontology about cars, their prices and sales regions and will give an insight into the ontology design process. It will get you in touch with the essential functions of NeOn Toolkit.

If you do not want to create the basic elements you can load a prebuilt demo ontology `Demo_basic_model_cars_en.oxml` that you can find in the installation folder under `examples\oxml`.

6.1. First Steps

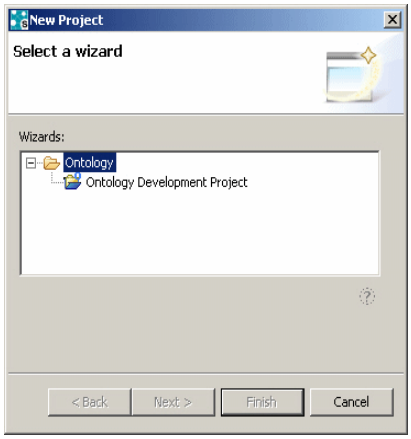
Before starting with the actual modelling we must set up the environment properly:

- Start NeOn Toolkit with a click on the desktop icon or via the Windows start menu.
- Ontologies always reside within a so called *ontology-project*. To set up a new project select ***File > New > Project …*** from the menu-bar.



Creating a new project

- In the „New Project“ window select ***Ontology*** and then ***Ontology Development Project.***



Creating a new Ontology Development Project

- Clicking on the ***Next*** button opens the window to create a new ontology project.
- Here enter the name „cars“ as the project name. You may of course enter an alternative name or the default one. This example will use the project name „cars“!
- To create the project click on the ***Finish*** button.

Create a new project named "cars"

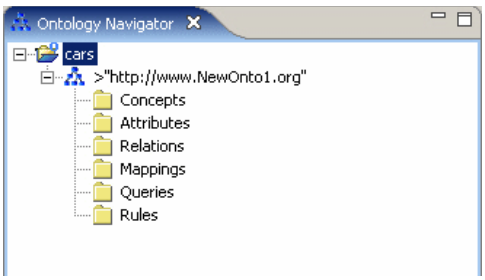
Before you can import or create an ontology you have to create a project, especially if NeOn Toolkit has been installed anew.

The „Ontology Navigator" now shows the new ontology project Next, we will demonstrate how to set up an ontology in this project.

6.2. Creating a New Ontology

In order to create a new ontology in NeOn Toolkit proceed as follows:

- Using the right mouse button, click on the project name in the „Ontology Navigator" and in the context menu select ***New Ontology***.
- In the „New Ontology" window you can now indicate a „Module Identifier" (the *name of the ontology*) and a „Default Namespace". The proposed default values are sufficient for the purposes of our sample. You can change them now if you want.
- The „Module Identifier" is used for unique identification of modules. Modules divide the knowledge base in smaller units to ensure that the units do not influence each other unintendedly. Thus, the module is equivalent to the notion of an ontology.
- After clicking on the „Finish" button you have now created a new, empty ontology. Note that the ontology is now displayed in the ontology navigator with empty folders for concepts, attributes, rules etc.



Basic structure of the ontology

2007 © Copyright lies with the respective authors and their institutions.

6.3. Creating New Concepts

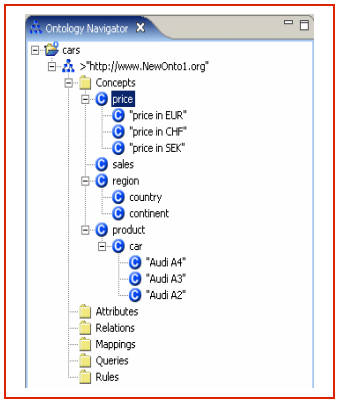
The next step is creating the concept „product“ as well as its sub-concept „car“ and the sub-concepts of „car“: „Audi A2“, „Audi A3“ and „Audi A4“.

- Open the entire structure of the ontology (click on the plus signs of the ontology) in the „Ontology Navigator“ window.
- With the right mouse button, click on the „Concepts“ folder and select „New Concept“ in the context menu.
- In the „OntologyNavigator“ window the default name of the concept, e.g. „term1103116866533“ appears within the „Concepts“ folder. Now change this name to a proper name, such as „product“.

Later modifications of the concept name are possible at any time using the Entity Property View (the top right area on the screen).

- With the right mouse button, click on the concept „product“ and from the context menu select „New Concept“.
- A new concept with a default name is now located under the „product“ concept. This indicates that the new concept is a sub-concept of „product“. Change the concept name to „car“.
- With the right mouse button, click on the concept „car“ and from the context menu select „New Concept“.
- Another new concept with a default name now appears under the „car“ concept, i.e. it represents a sub-concept of car. Change the concept name to „Audi A2“.
- Repeat these steps in order to create concepts „Audi A3“ and „Audi A4“ as sub-concepts of „car“.
- Whenever a word contains an umlaut (e.g. the “ü” in German „Reisef**ü**hrer“), a space character (such as in „Audi A4“), or any other non-word character the concept name will be enclosed in quotation marks.
- Next, we create the concepts „region“, „sales“ and „price“ as root concepts:
  - With the right mouse button click on the „Concepts“ folder and select „New Concept“ from the context menu.
  - Enter „region“ as the concept name and repeat this process also for „sales“ and „price“.
  - Finally create „continent“ and „country“ as sub-concepts of the „region“ concept.

After this process the structure of the ontology in the „Ontology Navigator“ should look like this (for better readability, the namespaces are deactivated):



Concepts and sub concepts

6.4. Multilingual Representations

Each concept can have representations in multiple languages. To create a German representation for the concept names enter them in the appropriate column of the „Representations“ table of the „Entity Property View“ of the individual concepts.

German representation „Produkt“ for the concept „product”

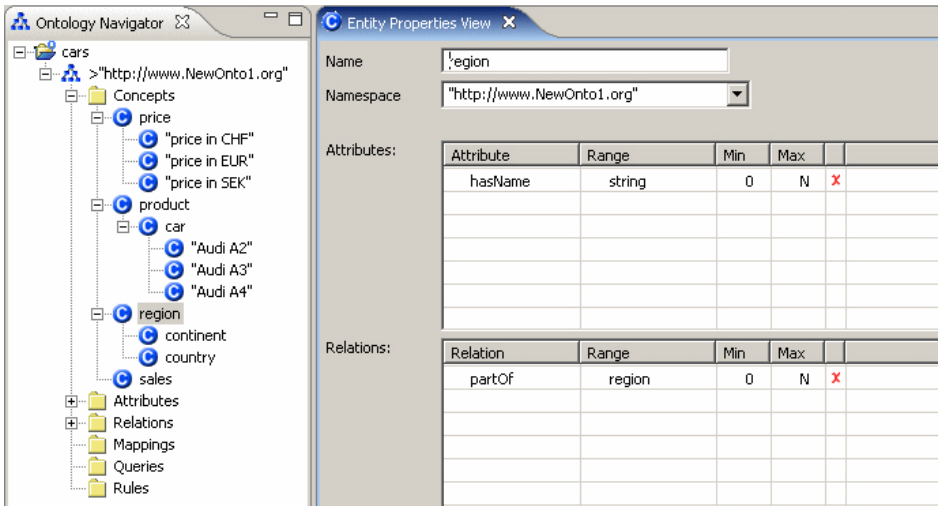
„product“	-> „Produkt“		„car“	-> „Auto“
„region“	-> „Region“		„continent“	-> „Kontinent“
„sales“	-> „Vertrieb“		„country“	-> „Land“
„price“	-> „Preis“			

6.5. Creating Attributes

The main distinction between attributes and relations is the type of their values. Attributes always have atomic values and thus are identical to OWL-data type properties. Relations always refer to instances and thus are equivalent to OWL-object properties.

Attributes and relations are defined for concepts in an object-oriented sense. They are, therefore, created and modified in the context of a concept within *the concept's* „Entity Property View“.

2007 © Copyright lies with the respective authors and their institutions.



Properties (attributes and relations) of the concept "region"

First we want to define an attribute "hasName" for products. In order to set up this attribute proceed as follows:

- Enter "hasName" in the „Entity Property View" of the "product" concept in the "Attribute" column under the section "Attributes".
- Enter "string" in the properties window of the "product" concept in the "Range" column under the section "Attributes" (or just type "s" and select "string" from the appearing list).
- Thus the property "hasName" is assigned to the concept "product" (and thereby all sub-concepts such as "car"). The changes are automatically stored which means that the new attribute also appears in the "Attributes" folder of the "Ontology Navigator".
- To verify that the attribute is inherited by "car" click on the "car" concept to open its „Entity Property View". Here we see the same definition of "hasName" as for "product". But now it has a grey background and cannot be modified.

To practice the creation of attributes you could define this knowledge:

- Region: hasName=>string
- Sales: amount=>integer
- Price: value=>number

6.6. Creating Relations

The connections between instances are represented with "relations". They are quite similar to attributes and are essentially created and maintained in exactly the same way. The first example we create is the relationship between regions. Each region can be part of another region, e.g. Germany is part of Europe.

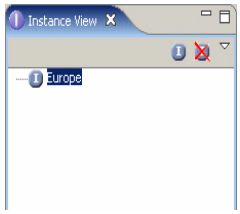
- Select the "region" concept in the "Ontology Navigator".
- In the „Entity Property View" of "region" enter "partOf" in the column "Relation" under the section "Relations". Choose "region" as the range of the relation. The changes are stored automatically and now the new relation appears in the "Relations" folder of the "OntologyNavigator".
- Relations are also inherited by sub-concepts. Verify this by checking the sub-concepts "continent" and "country".

To practice the creation of relations you could define this knowledge:

- Region: partOf=>region
- Sales: ofProduct=>product; inRegion=>region
- Price: ofProduct=>product; inRegion=>region

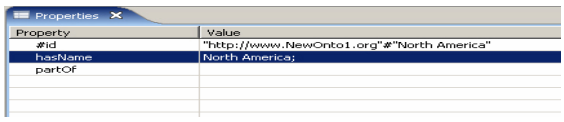
6.7. Setting up Instances

Using the above steps we have now created an ontology, i.e. a concept hierarchy with logical connections. Instances are concrete entities of concepts, e.g. “Europe” is an instance of the concept “continent”. To create instances we use the “Instance View”. It is shown below the “Ontology Navigator” window.



“Instance View” window

- First choose the concept you want to instantiate in the “Ontology Navigator”, here: the concept “continent”.
- Click on the symbol “Create a new instance” in the header of the “Instance View” to create a new instance. A new instance is created.
- An arbitrary name is suggested and you can change it immediately. Enter “Europe” as the name of the instance.
- In the “Properties” window to the right all potential properties of the instance are listed. Here you can fill in the correct value for the attribute field “hasName”.
- Repeat these steps to set up more instances of the concepts “continent” and “country”.



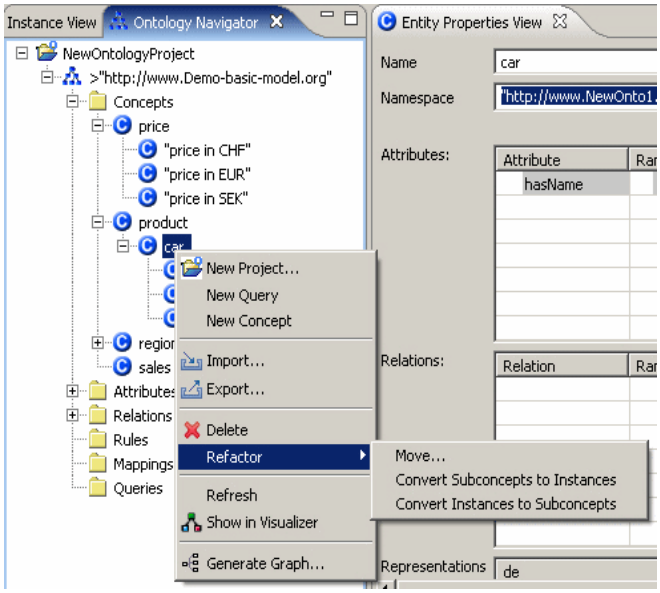
“Properties” window, e.g. the value “North America” in the attribute field “hasName”

- The “Properties” window also allows defining relations between instances, e.g. the above mentioned part of relation between countries and continents. Select for all defined countries the proper continents.

6.8. Moving a Concept

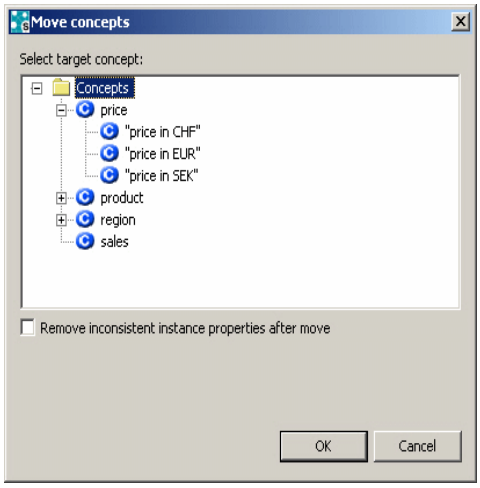
To rearrange the concepts in the concept hierarchy you can move the concepts around. After a right-click on the concept you can choose the “Refactor” menu to move an element.

2007 © Copyright lies with the respective authors and their institutions.



Menu – Moving a concept

With a click on the “Move…” instruction the window where you select the target concept will open. Alternatively, you can drag and drop a concept to its new parent (super-concept or concept folder).



“Move concepts” dialog to choose the target concept

The check box lets you choose whether inconsistent instance properties should be removed. When moving a concept, all its instances also move with it in the concept tree. Since instances carry real values (including values for inherited properties) some of them might not be valid after the move. By ticking the check-box all obsolete property values will be removed, otherwise they are retained, which actually represents inconsistent instances with respect to the information from the ontology.